

Prototipo de sistema inteligente basado en patrones de ondas cerebrales para prevenir accidentes de tránsito

Intelligent system prototype based on brainwave patterns to prevent traffic accidents

Amelia M. Brugiati^{1*}, Marvin X. González¹, Denzel H. Cornejo¹, Cristian I. Pinzón²

¹Licenciatura en Ingeniería de Sistemas y Computación, Centro Regional de Veraguas – Universidad Tecnológica de Panamá,

²Facultad de Ingeniería en Sistemas Computacionales, Centro Regional de Veraguas – Universidad Tecnológica de Panamá

Resumen Este artículo presenta el prototipo de sistema inteligente basado en patrones de ondas cerebrales para prevenir accidentes de tránsito, que, mediante un sensor colocado en la cabeza del conductor, monitoriza los patrones de ondas cerebrales los cuales son enviados en tiempo real vía Bluetooth a una placa Raspberry Pi para ser procesados con estrategias de aprendizaje automático y de esta forma enviar una alerta visual y sonora cuando detecta el estado de somnolencia en el conductor. Para la construcción del prototipo se recogieron datos de cuatro personas en tres estados distintos, mientras estaban despiertas, somnolientas y dormidas. El conjunto de datos fue procesado con cuatro algoritmos de aprendizaje supervisado: vecinos más cercanos, máquina de soporte vectorial, árboles de decisión, bosques aleatorios; siendo este último el que mejores resultados mostró alcanzando un 82.05% de precisión al diferenciar los tres estados anteriormente mencionados. El costo estimado del sistema es de 210 USD, resultando un sistema económico con relación a otros existentes en el mercado.

Palabras clave Accidentes de tránsito, bitalino, sistema inteligente, inteligencia artificial, aprendizaje automático, ondas cerebrales, Raspberry Pi, somnolencia.

Abstract This article presents the prototype of an intelligent system based on patterns of brain waves to prevent traffic accidents, by which, through a sensor, placed on the driver's head, monitors the patterns of brain waves that are sent in real time via Bluetooth to a Raspberry Pi to be processed with machine learning strategies. In this way it allows to send a visual and sound warning when it detects the state of drowsiness in the driver. For the prototype construction, data of four people were collected while they were awake, drowsy and asleep. The data set was processed with four supervised learning algorithms: nearest neighbors, support vector machine, decision trees and random forests; the last one was the one that obtained the best result, reaching 82.05% accuracy when differentiating the three different states. The estimated cost of the system is 210 USD, resulting an economic system in relation to others existing in the market.

Keywords Traffic accidents, bitalino, intelligent system, artificial intelligence, machine learning, brain waves, Raspberry Pi, drowsiness.

* Corresponding author: abrugiat97@gmail.com

1. Introducción

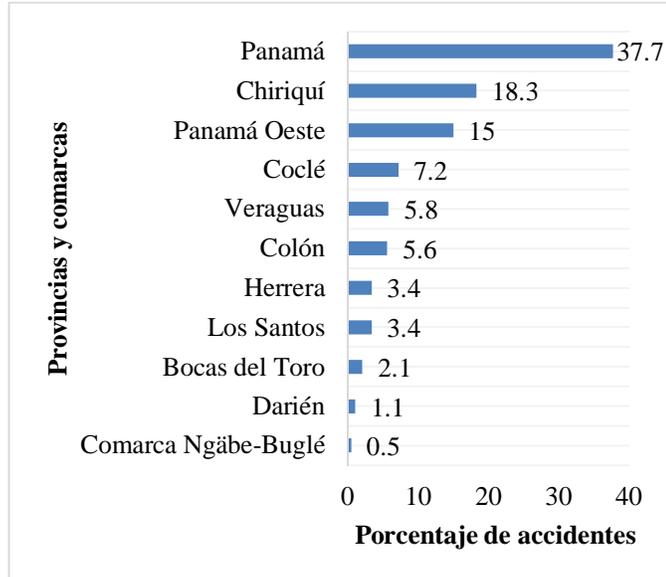
La demanda de transporte y de movimiento de las personas constituyen necesidades básicas para el desarrollo económico de un país. Sin embargo, entre más tiempo permanece una persona al volante mayor, es el riesgo de disminuir su concentración y reacción debido al cansancio y sueño, atentando contra la vida del conductor, pasajeros y peatones.

La somnolencia ha sido una de las principales causas de accidentes en países como Estados Unidos, donde según la estimación de la Administración Nacional de Seguridad del

Tráfico en las Carreteras (NHTSA), casi 56000 choques en el año 2006 estuvieron relacionados con el estado de somnolencia al conducir, representando el 20% del total de accidentes [1].

En Panamá, para el año 2017 datos arrojados por el Instituto Nacional de Estadística y Censo (INEC) indicaron que se registraron 56847 accidentes de tránsito, reflejando un aumento del 2.5% respecto al año anterior [2], lo que significa que alrededor de cada 13 minutos ocurre un accidente en distintos puntos del país, concentrándose en su mayoría en la provincia de Panamá (Gráfica 1).

Aunque no se lleven registros estadísticos de las causas de accidentes de tránsito en el país, diariamente se observan en los titulares nacionales accidentes causados por la somnolencia al volante [3], [4].



Gráfica 1. Porcentaje de víctimas en accidentes de tránsito en la república de Panamá, según provincia y comarca indígena: año 2017. Fuente: INEC.

Sobre la base de datos estadísticos alarmantes, en cuanto al número de accidentes de tránsito, se plantea la siguiente pregunta de investigación. Utilizando la tecnología existente, ¿Cómo se pueden monitorizar los patrones de ondas cerebrales de un conductor para determinar el estado de somnolencia y alertarlo cuando se esté durmiendo?

Las secciones en las que se divide la investigación son: contexto, materiales y métodos, resultados y discusión, trabajo futuro y conclusiones.

2. Contexto

La somnolencia [5] se puede considerar como un estado de transición entre el sueño y la vigilia, que se manifiesta como la falta de atención y deterioro del rendimiento mental o físico. Aquellos conductores que manejan en un estado de somnolencia tienen una mayor probabilidad de sufrir un accidente comparado con individuos que se encuentran en un estado de alerta o despiertos [6]. Por lo tanto, la detección temprana de la somnolencia es esencial para la prevención de accidentes.

Debido a los antecedentes mencionados se han desarrollado distintos tipos de proyectos para la detección temprana de la somnolencia en los conductores, utilizando principalmente sensores no invasivos.

2.1 Sensores basados en características visuales

La mayoría de las investigaciones realizadas en esta área se basan en el reconocimiento facial utilizando técnicas de inteligencia artificial para seguir la posición de la cabeza de los conductores y detectar la frecuencia de parpadeos o expresiones faciales [7]. Otras líneas de investigación se han centrado en reconocer patrones peligrosos de conducción como los cambios de carril y zigzag, utilizando sensores de giro dentro del vehículo [8].

2.2 Sensores basados en características no visuales

Las nuevas tecnologías proporcionan nuevos equipos, entre estos, *hardware* que permite medir distintos aspectos de la fisiología humana como la actividad neuronal, movimientos oculares, diámetro de la pupila, actividad cardíaca, temperatura corporal, entre otros [9].

Una de las técnicas que se pueden utilizar en la detección del comportamiento de conducción somnolienta es la electroencefalografía [1], la cual de manera no invasiva registra la actividad eléctrica del cerebro, generando uno de los indicadores más potentes para detectar el comportamiento de un conductor.

Estas señales del cerebro por sí solas no brindan información relevante, para esto deben ser analizadas para encontrar distintos patrones de comportamiento. En la actualidad, se ha buscado apoyo en técnicas de inteligencia artificial (IA) para el análisis de estas, en especial en el área de aprendizaje automático (ML, por sus siglas en inglés).

El aprendizaje automático es una rama de la IA, la cual aplica distintos algoritmos para establecer relaciones entre un gran conjunto de datos y generar información. Estos conjuntos de datos abarcan un dominio de problema existente que los distintos algoritmos ya sean de aprendizaje supervisado o no supervisado [10] utilizan para construir un modelo general, permitiendo que se generen nuevas predicciones con cierto grado de precisión en nuevos casos.

El objetivo principal de este trabajo de investigación es el diseño y construcción de un prototipo de sistema inteligente capaz de monitorizar las ondas cerebrales de los conductores y alertar en caso de riesgo de sueño al volante.

3. Materiales y métodos

3.1 Selección de las herramientas

Para la construcción del sistema inteligente se utilizaron distintos componentes tanto de *hardware* como *software*; en los siguientes puntos se describen los mismos.

3.1.1 Bitalino

La placa Bitalino [11] es un pequeño dispositivo de bajo costo el cual integra un conjunto de sensores para la adquisición de bioseñales. Es capaz de transmitir los datos

recogidos a distintos dispositivos en tiempo real mediante conexiones inalámbricas (figura 2a). Para la selección de este dispositivo se comparó con tres dispositivos que fueran capaces captar valores relacionados con la actividad cerebral (tabla 1). Entre estos se encontraban *Mindwave* de *Neurosky* [12], *Open BCI* [13] y *Emotiv Epoc* [14], resultando seleccionada la placa *Bitalino* principalmente por su disponibilidad, versatilidad y bajo precio.

• **Sensor EEG**

Dentro del conjunto de sensores que contiene la placa *Bitalino* se encuentra el sensor de electroencefalografía o EEG (figura 1). Este sensor utiliza una configuración bipolar para ayudar a eliminar el ruido de las regiones musculares donde se coloquen los electrodos [15]. Para poder utilizar este sensor se necesitan tres electrodos: uno positivo encargado de captar la señal, uno negativo para ayudar a la eliminación del ruido y uno de referencia que debe ser colocado en una región no muscular. Su funcionamiento se basa en detectar el potencial eléctrico del electrodo positivo y compararlo con el de referencia. El resultado es un valor análogo que va desde 0 a 1023, llegando a captar un máximo de 1000 valores por segundo. Estos valores son almacenados por la placa *Bitalino* y enviados a la placa *Raspberry Pi* para que sean procesados.

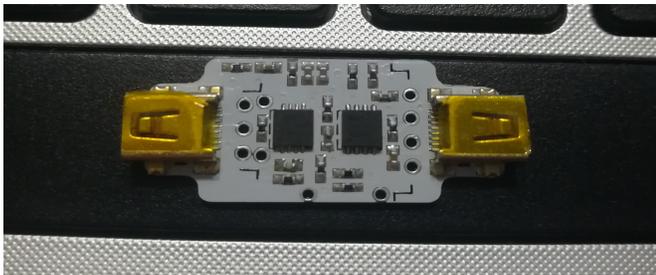


Figura 1. Sensor EEG del conjunto de sensores de la placa *Bitalino*.

3.1.2 Raspberry Pi

Raspberry Pi es un computador pequeño y de bajo costo (50.00 USD) (figura 2b). La placa cuenta con el sistema operativo de *software* libre *Raspbian* [16] para su funcionamiento, la misma tiene una gran capacidad para compilar y ejecutar programas escritos en distintos lenguajes [17]. Para el desarrollo de este proyecto se utilizó el modelo 3B ya que incluye conexión *bluetooth* facilitando la transferencia y recepción de datos.

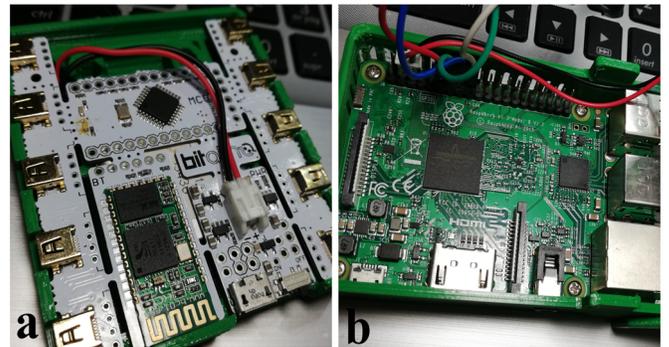


Figura 2. [a] Placa *Bitalino* para recolección de datos. [b] Placa *Raspberry Pi* utilizada para procesar datos recibidos por la placa *Bitalino*.

3.2 Participantes

Para obtener el conjunto de datos se seleccionaron cuatro personas entre estos hombres y mujeres entre las edades de 25 a 56 años. Ninguno de los participantes utilizó medicamentos ni consumieron alcohol, *cafeína* o alguna bebida energética durante la recolección de datos.

Tabla 1. Comparación de dispositivos enfocados en la adquisición de señales cerebrales

	Precio (USD)	Canales	Ventajas	Desventajas
BITalino	149.99	Canal Bipolar con frecuencias de muestreo de 1, 10, 100 y 1000 Hz.	Adquisición de señales EEG, EMG, ECG y EDA. API's de desarrollo completas.	Falta de información sobre librerías para procesar los datos.
Mindwave	99.99	Un solo canal monopolar ubicado en FP1 con una frecuencia de muestreo de 512 Hz.	Detección de parpadeos y nivel de concentración. Multiplataforma.	Escasez en el mercado.
Open BCI	499.99	8 canales con frecuencia de muestreo de 16 kHz.	Posibilidad de almacenar datos en una tarjeta MicroSD. Compatibilidad con <i>Arduino</i> .	Precio elevado.
Emotiv Epoc	399.99	14 canales con una frecuencia de muestreo de 2000 Hz.	Compatibilidad con sistemas móviles.	Datos cifrados y se debe pagar un precio adicional por obtención de señales en bruto.

3.3 Recolección de datos

A cada uno de los participantes se le tomaron datos en tres condiciones distintas, por cinco segundos en intervalos de un minuto, realizándose en total 15 mediciones por cada participante; cinco mediciones mientras conducían en horas de la tarde y noche, cinco cuando presentaban somnolencia en un estado de reposo y cinco cuando estaban completamente dormidos.

Los datos fueron recogidos mediante un conjunto de electrodos colocados en la cabeza de cada participante (figura. 3) y conectados a la placa Bitalino. Estos datos sin procesar se enviaron a la placa Raspberry Pi que se encargó de procesarlos y segmentarlos mediante un código escrito en el lenguaje de programación Python [17]. Este código utilizó un módulo denominado *Biosppy* (Procesamiento de bioseñales en Python) [18], el cual clasificaba los datos en cinco características distintas.



Figura 3. Recolección de datos en conductores utilizando la placa Bitalino.

A los datos obtenidos se les asignó una clase manualmente dependiendo el estado del conductor (tabla 2). En total se obtuvo una matriz de datos de 2341 filas por cinco columnas. Estos datos se guardaron en un archivo CSV (del inglés *comma-separated values*) para poder ser procesados.

Tabla 2. Extracto del conjunto de datos procesados por la librería *Biosppy*, donde las cinco primeras columnas representan las características extraídas y la última representa la clase a la cual pertenece

theta	alpha_low	alpha_high	beta	gamma	clase
6.37	6.79	16.39	96.30	23.07	despierto
12.02	17.69	26.46	96.30	23.07	despierto
5.87	6.62	3.57	5.07	1.46	somnoliento
5.87	10.04	2.56	5.07	1.61	somnoliento
0.45	2.19	1.89	0.93	0.45	dormido
1.31	2.19	2.42	1.58	0.46	dormido

3.4 Entrenamiento del sistema

El objetivo principal del aprendizaje automático es generar predicciones de un nuevo conjunto de datos a partir de un modelo previamente entrenado. Teniendo el conjunto de datos clasificados en cada clase (despierto, somnoliento y dormido)

se separaron en datos de entrenamiento y prueba. Los datos de entrenamiento alimentaron los algoritmos para la extracción de conocimiento y el conjunto de datos de prueba evaluó la precisión del algoritmo.

Para la construcción del modelo se seleccionaron cuatro algoritmos de aprendizaje supervisado [10] y se utilizó la librería *Sklearn* de Python [18] para la implementación de estos, que con su conjunto de herramientas permitió el análisis de los datos obtenidos; estos algoritmos fueron: vecinos más cercanos (KNN) [19], máquina de soporte vectorial (SVM) [20], árboles de decisión [21] y bosques aleatorios (RF) [22].

En el caso del algoritmo de KNN se construyó un programa para identificar el valor de K con el que se obtenía una mejor precisión. Además, para conocer el rendimiento de cada algoritmo se construyó una matriz de confusión [23] para visualizar con que precisión clasificaba cada uno de los datos en las tres clases definidas.

3.5 Construcción del prototipo

Teniendo el sistema entrenado con el algoritmo que obtuvo la mejor precisión se procedió a construir el prototipo el cual funciona de la siguiente manera: la placa Bitalino capta los datos recibidos por los electrodos colocados en la cabeza del conductor (figura 4) y por medio de una conexión Bluetooth los envía a la placa Raspberry Pi conectada a una salida tipo USB de cinco voltios (5v) del auto; luego de captados los datos, son procesados y llevado al algoritmo previamente entrenado que clasifica la clase de la nueva muestra.

Dependiendo de su clasificación, se envía una señal visual (verde=despierto, amarillo=somnoliento o rojo=dormido) mediante un led RGB conectado a los pines GPIO [24] de la placa Raspberry Pi; además, si el sistema detecta el estado “dormido” en el conductor, envía también una alerta sonora por medio del conector de audio de la placa. Este proceso se realiza en intervalos de un minuto indefinidamente hasta que la placa se desconecte.

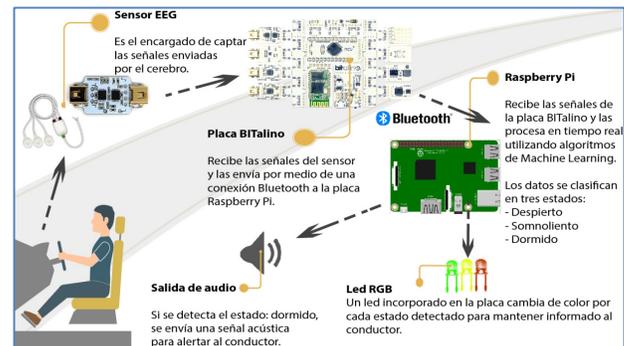
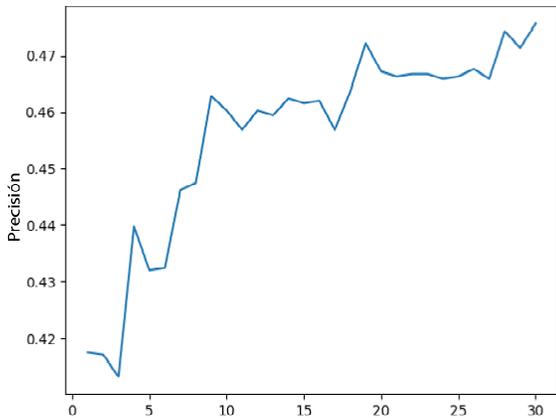


Figura 4. Diseño propuesto del prototipo con sus componentes.

4 Resultados y discusión

En el caso del algoritmo KNN se procesaron los datos con distintos valores para K (gráfica 2) para seleccionar el que mejor precisión mostraba. En base a los datos generados se

construyó el modelo, el cual obtuvo una precisión máxima de 67.52%, por lo cual se descartó su uso.



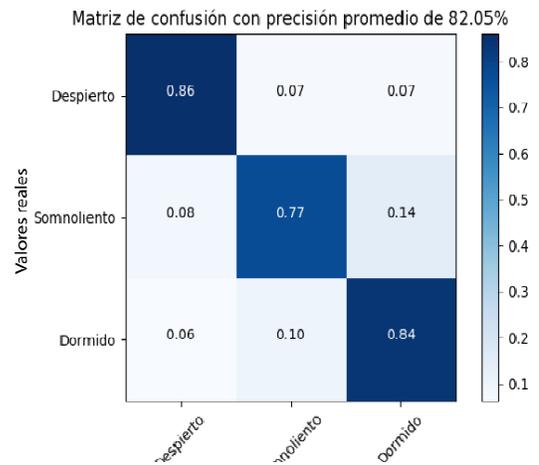
Gráfica 2. Valores de K para algoritmo de KNN donde el eje de las x representa el valor de K y el eje y la precisión obtenida en cada iteración.

En la tabla 3 se observan los algoritmos utilizados y la precisión para clasificar los datos de cada uno, siendo el algoritmo de bosques aleatorios el que mejores resultados obtuvo, alcanzando una precisión del 82.05%.

Tabla 3. Algoritmos de aprendizaje supervisado utilizados con la precisión máxima obtenida

Algoritmo utilizado		Precisión
Vecinos más cercanos con K=15		47.33%
SVM	Lineal	46.29%
	Gaussiano	64.11%
Árbol de decisión		67.52%
Bosques aleatorios		82.05%

Al ser bosques aleatorios el que mejor resultado presentó (82.05%), se construyó una matriz de confusión para visualizar el comportamiento de los datos (gráfica 3) utilizando este algoritmo y se pudo observar que presentaba un nivel más bajo de precisión cuando diferenciaba entre los estados de somnoliento de dormido, bajando hasta el 77%. Sin embargo, mostraba niveles altos de para clasificar los estados despierto y dormido.



Gráfica: el eje x representa las clases reales a las que pertenecen los datos y el eje y las predicciones realizadas por el algoritmo de bosques aleatorios para nuevas muestras.

5 Trabajo futuro

Para mejorar los resultados y confiabilidad del prototipo se pretende utilizar los sensores adicionales de la placa Bitalino, para procesar otras bioseñales, como son: electromiografía (EMG) y electrocardiografía (ECG) de manera simultánea.

Además, en conjunto a las ondas cerebrales monitorizar los rasgos faciales del conductor, haciendo uso de técnicas de procesamiento de imágenes.

Por otra parte, utilizar otros sensores existentes en el mercado para evaluar el rendimiento comparado con la placa Bitalino.

En relación con esta última placa mencionada, se puede diseñar un modelo en forma de casco donde puedan ubicarse los cables que esta tiene para disminuir la molestia que estos puedan causar mientras se conduce.

6 Conclusiones

A nivel mundial los accidentes de tránsito provocados por la somnolencia generan estadísticas significativas cada año y Panamá no escapa de esta realidad.

Utilizando la tecnología disponible en el mercado se pueden desarrollar proyectos enfocados a la prevención de accidentes de tránsito y los cambios en las ondas cerebrales de las personas son una forma para generar alertas mientras se conduce.

En la realización del proyecto se identificaron las tecnologías *hardware* y *software* necesarias para la lectura y procesamiento de ondas cerebrales y en base a estas se diseñó el prototipo de sistema inteligente para prevenir accidentes de tránsito utilizando estrategias de aprendizaje automático.

La implementación del algoritmo de bosques aleatorios arrojó mejores resultados en el nivel de precisión para predecir nuevas muestras, en comparación a los otros tres utilizados.

El prototipo construido alcanzó un nivel de precisión de un 82.05%, para predecir una nueva muestra en los estados de un conductor despierto, somnoliento y dormido.

Los resultados se pueden mejorar cambiando los parámetros de los algoritmos de ML.

La monitorización de ondas cerebrales que presenten variaciones mientras se conduce, pueden ayudar a la detección temprana del cansancio, lo cual, con la debida reacción del sistema, puede alertar al conductor y así prevenir un posible accidente.

AGRADECIMIENTOS

Agradecemos a la profesora María Luisa Vélez por apoyarnos en la adquisición de las tecnologías y en la realización del proyecto. Al doctor José Carlos Rangel por la orientación referente a temas de *machine learning*.

REFERENCIAS

- [1] M. S. Wang *et al.*, “Drowsy behavior detection based on driving information,” *Int. J. Automot. Technol.*, vol. 17, no. 1, pp. 165–173, 2016.
- [2] Instituto Nacional de Estadística y Censo, “Comentario,” 2018. [Online]. Available: <https://www.contraloria.gob.pa/inec/archivos/P8851CO-MENTARIOS-2017.pdf>. [Accessed: 16-Sep-2018].
- [3] Redacción web, “Choca y vuelca vehículo en la vía Transistmica, tras quedarse dormido al volante,” *Crítica Panamá*, Panamá, 03-Jan-2018.
- [4] V. E. Rodríguez, “Hombre pierde la vida al colisionar con un árbol en Santiago,” *Panamá América*, Panamá, 05-Aug-2018.
- [5] E. Rosales Mayor and J. R. De Castro Mujica, “Somnolencia: Qué es, qué la causa y cómo se mide,” *Acta Médica Peru.*, vol. 27, no. 2, pp. 137–143, 2010.
- [6] J. Vicente, P. Laguna, A. Bartra, and R. Bailón, “Drowsiness detection using heart rate variability,” *Med. Biol. Eng. Comput.*, vol. 54, pp. 927–937, 2016.
- [7] W.-B. Horng and C.-Y. Chen, “A Real-Time Driver Fatigue Detection System Based on Eye Tracking and Dynamic Template Matching,” *Tamkang J. Sci. Eng.*, vol. 11, no. 1, pp. 65–72, 2008.
- [8] E. Jeong, C. Oh, and I. Kim, “Detection of Lateral Hazardous Driving Events using In-vehicle Gyro Sensor Data,” *KSCE J. Civ. Eng.*, vol. 17, no. 6, pp. 1471–1479, 2013.
- [9] M. Kandemir, “Learning Mental States from Biosignals,” Aalto University, 2013.
- [10] M. Awad and R. Khanna, “Machine Learning,” in *Machine Learning*. In: *Efficient Learning Machines*, California: Apress, Berkeley, CA, 2015, pp. 1–18.
- [11] H. P. Da Silva, A. Lourenço, A. Fred, and R. Martins, “BIT: Biosignal Igniter Toolkit,” *Comput. Methods Programs Biomed.*, vol. 115, no. 1, pp. 20–32, 2014.
- [12] NeuroSky, “MindWave Mobile,” 2004. [Online]. Available: <http://neurosky.com/>.
- [13] OpenBCI, “OpenBCI,” 2015. [Online]. Available: <http://openbci.com/>.
- [14] Emotiv, “EMOTIV EPOC,” 2011. [Online]. Available: <https://www.emotiv.com/>.
- [15] Biosignalsplux, “Electroencephalography (EEG) Sensor Data Sheet,” 2015. [Online]. Available: https://biosignalsplux.com/datasheets/EEG_Sensor_Data-sheet.pdf. [Accessed: 24-Sep-2018].
- [16] Raspberry Pi Foundation, “Installing operating system images - Raspberry Pi Documentation.” [Online]. Available: <https://www.raspberrypi.org/documentation/installation/installing-images/README.md>. [Accessed: 13-Jun-2018].
- [17] M. Richardson and S. Wallace, *Getting Started with Raspberry Pi*. Maker Media, Inc, 2013.
- [18] R. Garreta and G. Moncecchi, *Scikit-learn: Machine Learning in Python*, vol. 12. 2013.
- [19] R. S. Zack, C. Tappert, and S.-H. Cha, “Performance of a long-text-input keystroke biometric authentication system using an improved k-nearest-neighbor classification method.” 2010.
- [20] T. Yu, J. Debenham, T. Jan, and S. Simoff, “Combine Vector Quantization and Support Vector Machine for Imbalanced Datasets,” in *Artificial Intelligence in Theory and Practice*, Springer US, 2006, pp. 81–88.
- [21] T. Keck, “FastBDT: A Speed-Optimized Multivariate Classification Algorithm for the Belle II Experiment,” *Comput. Softw. Big Sci.*, vol. 1, no. 1, p. 2, Nov. 2017.
- [22] Y. Zhou and G. Qiu, “Random Forest for Label Ranking,” vol. 112. 2018.
- [23] scikit-learn developers, “scikit-learn user guide Release 0.19.2,” 2018.
- [24] Paragon Internet Group Ltd, “Raspberry Pi Pinout,” 2015. [Online]. Available: <https://pinout.xyz>. [Accessed: 17-Jun-2018].