

PROCESAMIENTO EN BORDE EN INTERNET DE LAS COSAS PARA TRANSMISIÓN DE INFORMACIÓN DE INTERFACES CEREBRO COMPUTADORA

Juan A. Ramírez Quintana¹, Jaime Trujillo Garcia², Mario I. Chacón Murguía¹
Laboratorio de Percepción Visual con Aplicaciones en Robótica¹, Laboratorio de innovación de Intel²
Tecnológico Nacional de México/I.T. Chihuahua
Ave. Tecnológico No. 2909, 31310, Chihuahua, Chih., México, (614)2026511
jaramirez@itch.edu.mx, jamesoner07@gmail.com, mchacon@ieee.org

RESUMEN.

Las interfaces cerebro computadora (BCI por sus siglas en inglés) son una tecnología útil para apoyar el manejo de las actividades cotidianas de personas con discapacidad. Una tendencia reciente en el desarrollo de los BCI es el acceso de esta tecnología al Internet de las cosas (IoT) para el desarrollo de prótesis, sistemas de control con aplicaciones domóticas, seguimiento médico, etc. En este artículo se desarrolla una red de procesamiento al borde para sistemas BCI que procesan señales electroencefalográficas P300 con el objetivo de compartir información de las ondas P300 a diversas plataformas conectadas a dicha red pero sin la necesidad de tener acceso a la nube. Como resultados se obtuvo sistema en borde con servidor centralizado con un tiempo de procesamiento de 1 ms por muestra y una detección de estímulos con una precisión del 93.75%.

Palabras Clave: interfaz cerebro computadora, internet de las cosas, electroencefalografía, procesamiento en borde.

ABSTRACT.

Brain-computer interfaces (BCI) have been a useful technology to support people with disabilities in daily activities. A new trend in the development of BCI technology is the access of these interfaces to the Internet of Things (IoT) in applications like prostheses, control systems in home automation applications, medical monitoring, etc. In this article, an edge processing network is developed for BCI systems that processes electroencephalographic P300 signals in order to share information from P300 signals to several platforms connected to network but without the use of the Internet cloud. As results, an edge system with a centralized server is obtained with a processing time of 1 ms per sample and a detection of stimuli with an accuracy of 93.75%.

Keywords: brain computer interface, internet of things, electroencephalography, edge processing

1. INTRODUCCIÓN

Las interfaces cerebro computadora (BCI) es una tecnología que auxilia a pacientes con discapacidad en la realización de diversas tareas cotidianas [1]. Las aplicaciones para BCI varían según las necesidades del paciente a tratar y cubren campos como el control de motores [2], movimiento en 2 dimensiones [3], detección de estímulos para comunicación [4], control de sillas de ruedas [5], control de movimiento con robots móviles, control de brazos robótico [6], etc.

Entre las aplicaciones para BCI, existen trabajos que integran esta tecnología con el Internet de las cosas (IoT) para potenciar las diversas cualidades de estas interfaces [7] [8]. IoT puede integrar el concepto de computación ubicua a cualquier aplicación con el objetivo de tener acceso directo y procesar la información bajo un sistema portable cuyos elementos no sean notorios [9] [10].

Como tal, existen diversos trabajos previos que integran estas dos tecnologías (IoT y BCI) orientados al control de sillas de ruedas con conexión a la nube [11], monitoreo con actuadores y sensores basados en deep learning con interacción en IoT [12], manejo de *big data* con algoritmos predictivos para BCI [13]. Dentro de los trabajos mencionados se consideran de manera global tres conceptos que son BCI, deep learning y conexión con la nube. El mayor inconveniente es la conexión con la nube, ya que no en todos los casos es requerido tener interacción directa con los servidores de la nube, por el contrario, el acceso a estos servidores genera altos costos y tiempos de latencia altos [14]. Para mejorar el desempeño de estos sistemas se puede recurrir a desarrollar una red IoT basada en procesamiento en borde que es una forma de implementar esquemas de computación ubicua. De manera general, este tipo de tecnología busca integrar al entorno del usuario múltiples tecnologías sin que pueda percibir cada una de ellas como una gran variedad de dispositivos [15].

Por lo tanto, en este artículo se propone realizar un sistema BCI que procesa señales electroencefalográficas (EEG) P300 basado en computación ubicua con procesamiento en borde para tener tiempos reducidos de procesamiento conectados a un servidor centralizado y con esto poder monitorear a pacientes a distancia mejorando velocidad, reduciendo costos de una manera eficiente. El uso de las ondas P300 es debido a que de acuerdo con la literatura científica, la señal EEG P300 es un potencial evocado estable utilizado en la comunicación humano máquina de personas con grados severos de discapacidad.

El resto de las secciones, tienen el siguiente contenido. En la sección 2 se presenta el desarrollo general del sistema, la sección 3 presenta los resultados generales y la sección 4 las conclusiones.

2. DESARROLLO DEL SISTEMA.

En la Figura 1 se puede ver un esquema general del método propuesto, el cual consta de 4 etapas. La primera es la adquisición y consiste en una interfaz cerebro computadora de uso libre llamada OpenBCI®. Esta interfaz incluye una diadema con sus respectivos electrodos y su tarjeta de adquisición de datos comunicada por medio de bluetooth a un receptor externo. La segunda etapa es el procesamiento de las señales y consiste en un sistema embebido que procesa las señales EEG para detectar la onda P300. Esta sistema embebido es una Raspberry Pi 3B y el método para el procesamiento es una red neuronal convolucional (CNN). En la tercera etapa, la información resultante del sistema embebido se almacena en un servidor centralizado para almacenamiento y posterior uso. Finalmente, la etapa final es un conjunto de servicios para analizar las señales implementados en computadoras y/o dispositivos móviles. Estos servicios pueden ser graficar las señales EEG, realizar entrenamientos de redes neuronales, etc.



Figura 1. Estructura general del sistema

2.1 Adquisición de datos.

La adquisición de las señales EEG se realiza con los electrodos posicionados en la diadema Ultracortex Mark IV con la tarjeta de adquisición Cython+Daisy de OpenBCI®. La diadema coloca los electrodos con base en el sistema internacional 10/20 y su capacidad 16 canales, pero con base en los trabajos de [16], se utilizaron para este trabajo los electrodos Oz, A1 y Fpz como se muestra en la Figura 2. Oz mide actividad EEG, mientras que A1 y Fpz son referencia y tierra. Los electrodos utilizados fueron de copa de oro y se utilizó la pasta conductora Ten20 para reducir impedancias. La tarjeta Cython+Daisy funcionan con una frecuencia de muestreo de 250 Hz con una resolución de 24 bits por muestra.

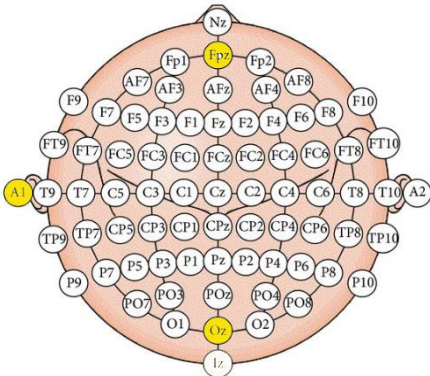


Figura 2. Posiciones de los electrodos según el sistema internacional 10/20.

Para evocar el P300, el paciente debe fijar su atención en un estímulo externo que aparece en forma aleatoria. Cada que aparece dicho estímulo, se debe reflejar la onda P300 en las señales EEG. Para este trabajo, el P300 se evoca mediante estímulos externos parpadeantes, para lo cual, se utilizó una interfaz visual que se muestra en la figura 3 y consta de un frutero y 4 cuadros que parpadean a diferentes tiempos en forma aleatoria (Estos cuadros se encuentran enmarcados en color gris en la figura 3). El parpadeo se realiza cambiando el color de cada cuadro negro a blanco. Cada cuadro porta un mensaje (si, no, arriba, abajo) y cuando el paciente necesita decir algo, solo debe fijar su mirada en el estímulo que contiene el mensaje. De esta manera, cuando se genera el P300 en la señal EEG, se puede determinar cuál estímulo está el paciente fijando su atención y en consecuencia saber el mensaje que pretende decir.



Figura 3. Interfaz para generar estímulos.

La tarjeta de adquisición funciona con una frecuencia de muestreo de 250Hz [16] y toma las señales EEG que pueden contener la onda P300 y la pasa a la Raspberry Pi ® mediante Bluetooth ®. Una vez que las señales llegan a la Raspberry, estas se van guardando las muestras en archivos separados por comas (.CSV).

2.2 Procesamiento con redes neuronales.

La siguiente etapa implementa el procesamiento de las señales a través de una Raspberry Pi conectada por medio de bluetooth al OpenBCI®. La Raspberry toma los datos almacenados en los archivos .CSV y aplica un filtro Butterworth pasa banda de 1-20 Hz [16]. Posteriormente, se sigue de una normalización de la siguiente forma:

$$S(n) = \sum_{i=0}^2 \frac{1}{\sqrt{\frac{\sum_{i=0}^2 (x_i(n) - \bar{x})^2}{N-1}}} (x_i(n) - \bar{x}) \quad (1)$$

Donde $x_i(n)$ es la señal actual, N es el total de muestras, i es el índice de cada electrodo ($i=\{O1,A2,Fpz\}$), \bar{x} es el promedio del total de la señal $x(n)$, $S(n)$ es normalización de la señal EEG y n es el identificador de tiempo en las señales que va de 0 a 149, lo cual se refiere a 600 ms que es la ventana de tiempo en que se da la P300.

Una vez realizado todo el procesamiento para adquirir y acondicionar las señales, $S_i(n)$ se aplica a una red neuronal convolucional como se implementa en [3] para realizar un modelo adaptativo al paciente con el que se esté tratando. El modelo de la red CNN se implementó en lenguaje de programación Python® con la librería TensorFlow® la cual da la facilidad de optimizar y entrenar la red con soporte para la Raspberry Pi®.

En la figura 4 se presenta un diagrama a bloques que representa cada capa del modelo de la red CNN implementada en el trabajo de [16], pero a continuación se dará un breve resumen de lo que entra y sale en cada capa de la red. La señal $S(n)$ entra en el bloque convolucional donde se le aplica un conjunto de 30 filtros de 1×70 resultando $Sc_{j,k}$, donde j es el índice del filtro, ($j=1, \dots, 29$) y k es el índice de un dato en el tiempo de la señal filtrada, ($k=0, \dots, 30$). Los datos en el tiempo pasaron de $n=0, \dots, 149$ a $k=0, \dots, 30$ debido a una reducción en el tiempo generada por los filtros. Posteriormente, se conecta con la capa de función de activación ReLU y la salida es $Sr_{j,k}$ la cual es de la misma dimensión que $Sc_{j,k}$. La siguiente capa es una totalmente conectada (*fully connected*) y reacomoda los datos de entrada para analizarlos y categorizarlos en cualquiera de dos clases de salida dando como resultado $Sfc_m(n)$ donde m representa al índice de cada clase ($m=\{P300, NoP300\}$), resultando en un vector de 2 valores que equivalen a cada clase. Una clase es representada por la probabilidad de que la onda P300 existe y la otra que la onda P300 no existe. De esta etapa. Las dos clases pasan por una capa de función *softmax* para conocer la probabilidad de pertenencia para ser clasificados al final y que la red determine si el estímulo existe o no, resultando en un vector de dos datos por señal de salida $Ss_m(n)$ de la misma dimensión que $Sfc_m(n)$.

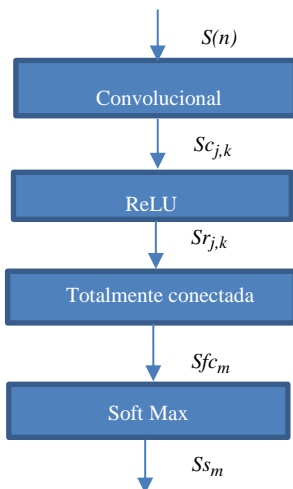


Figura 4. Modelo de red neuronal convolucional.

Cada capa tiene sus pesos y sus bias definidos en [16]. Para entrenar la red, la adquisición de las señales se basa en el protocolo señalado en [16]. Se toma un total de 192 señales en donde se tiene estímulo en cada una de ellas según el cuadro que el paciente esté observando. De estas 192 se utilizan 112 para el entrenamiento de la red CNN y el resto para comprobar los resultados. Posterior a realizar el entrenamiento y las pruebas de la red el modelo se guarda en archivos separados por comas para los pesos y bias de la red, para luego ser almacenados en el servidor central. El entrenamiento se realiza por paciente debido a que la detección de P300 demanda ajustar diferentes modelos matemáticos de manera individual a cada uno.

2.3 Almacenamiento en servidor central.

Como se mencionó previamente, hasta la sección anterior es la implementación del dispositivo que adquiere y analiza las señales EEG, los cuales se basaron en [16]. A estos dispositivos se les denominada BCI embebidos. Sin embargo, la contribución de este trabajo es la red IoT, la cual tiene un servidor centralizado que adquiere las señales de uno o varios dispositivos con la estructura descrita en las secciones 2.1 y 2.2. Es decir, como se ve en la figura 5, el sistema es escalable y puede recibir la información de varios dispositivos EEG. El servidor tiene como objetivo enviar la información de la detección del P300 y de la misma señal EEG a otros dispositivos que brindan información servicios a pacientes o médicos con propósitos de seguimiento clínico, mover prótesis, robótica, etc. Esto último se refiere al servicio de análisis, el cual se compone por computadoras, dispositivos móviles o microcontroladores.

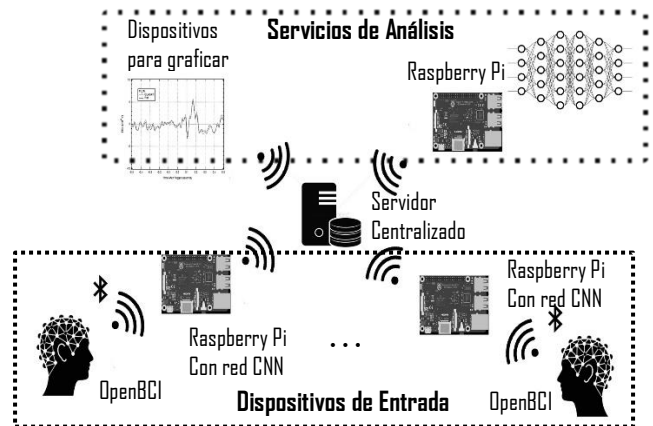


Figura 5. Ejemplo de dispositivos conectados al sistema.

Los BCI embebidos se conectan al servidor central Por medio del módulo WiFi integrado en la Raspberry Pi ®. Como el sistema es escalable, se pueden conectar al servidor central diversos BCI embebidos para adquirir o analizar la señal P300 y cada uno se le designa un nombre de usuario y contraseña, el cual solo puede ser dado de alta desde el servidor esto sin tener conexión a la nube. A estos usuarios se les puede configurar tiempos de acceso, cantidad de solicitudes por hora o simplemente denegar cualquier tipo de acción. Para acceder a este se le hace la solicitud desde cada dispositivo en donde se podrá almacenar información hasta que de manera automatizada este se lo permita.

El servidor central es una computadora con procesador Intel® i7-3770, 8 GB de memoria RAM y un adaptador wifi TP-Link® TL-WN722N. La conexión intermedia se hizo a un ruteador Linksys® WRT54G el puente del router a la Raspberry Pi ®. Para el almacenamiento de los datos en dicho servidor, se hace uso de una base de datos estructurada en SQL con modelos de entidad relación [17]. Este tipo de modelo permite configurar los datos por paciente como se muestra en la Figura 5. Para cada uno, se genera una tabla en la base de datos que contiene otras tres sub tablas cuya información es las señales EEG adquiridas, los pesos y bias de la red CNN, como se ve en la figura 6. Esta configuración de la base de datos se debe a que la red neuronal se debe entrenar por paciente, ya que las señales EEG son distintas en cada paciente.

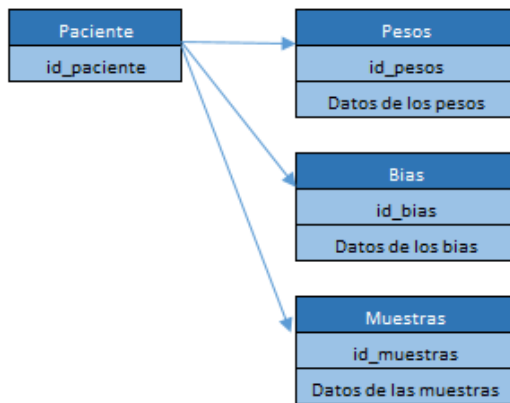


Figura 6. Modelo entidad relación

En el servidor se crea una tabla para cada paciente con el nombre o el número de identificación el cual siempre pertenecerá a este de manera única. Una vez creada la tabla principal se crean otras tres tablas que contienen los datos que se van a utilizar. Después de esto, se empieza a almacenar los datos en esas tablas. En caso de que el paciente ya exista y solo se requiera modificar los datos del paciente se debe usar su número de identificación para encontrar los datos.

De esta manera si la base de datos de los pacientes crece se mantendrá de manera organizada los datos para cada uno sin necesidad de alterar datos o números de identificación. Además

que se puede escalar agregando más dispositivos sin afectar al sistema.

Esta base de datos es alimentada por todos los BCI embebidos conectados y pone la información disponible para los dispositivos o computadora que realicen los servicios de análisis.

Un aspecto que se debe considerar en este servidor, es que no se hace acceso a los servidores de la nube, con lo cual, no existen tiempos de latencia por la conexión y permisos de acceso.

2.4 Servicios de análisis.

Como se puede observar en la Figura 5, el servicio de análisis se da por diferentes dispositivos que procesan la información del servidor central derivadas de los BCI embebidos. De esta manera, la red IoT pone a disposición la información EEG o los mensajes de un paciente con problemas severos de discapacidad a diferentes sistemas que pueden auxiliar a un paciente en sus actividades cotidianas, seguimiento médico o simplemente comunicarse con una persona.

Para el caso de la red propuesta, se implementaron dos servicios, un es de graficar la señal EEG en línea, lo cual consiste en obtener en tiempo real los datos de la señal EEG en una gráfica, como muestra en la figura 7. De esa manera se tiene una herramienta para posible estudio del comportamiento del paciente. El acceso a los datos del servidor centralizado presenta la posibilidad de procesar las señales en diversos dispositivos a la vez sin necesidad de tener de manera presencial a un especialista. La herramientas para desarrollo fue *matplotlib()* de Python.

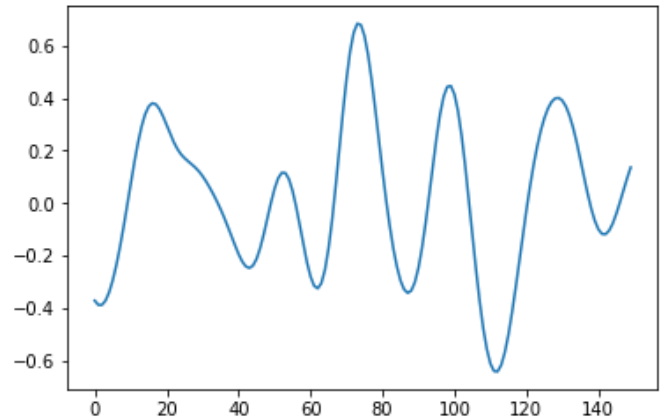


Figura 7. Grafica de una señal EEG.

Otro servicio que se implementó es el de entrenar la red CNN en otros sistemas de cómputo como en este caso fue una laptop para que en caso de que se anexe un paciente nuevo, los datos de pesos y bias se guarden desde cualquier dispositivo y poder realizar la sesión P300 desde cualquier parte.

Ambos servicios se implementaron en una computadora con procesador Intel i5 de segunda generación, 4 GB de RAM y adaptador wifi TP-Link® TL-WN722N.

El paciente y los servicios de análisis no tienen que estar físicamente, mientras ambos se encuentran dentro del servidor centralizado. Para configurar la red, una persona autorizada podrá dar de alta a los dispositivos desde el servidor definiendo estos parámetros. Esto para que el servidor pueda dar permiso, evitando que dispositivos no deseados accedan.

3 RESULTADOS

Esta sección se analiza el desempeño del sistema en velocidad y precisión para procesar y transmitir las señales EEG y el resultado de la detección de la onda P300. Para esto se realizaron pruebas con señales EEG las cuales constaron cada una de un vector con 150 muestras con tipo de dato flotante. Debido a que el enlace con el servidor se apoya de adaptadores WiFi la distancia es un factor tanto en velocidad como en desempeño para el manejo de los datos. Por lo que las mediciones se realizaron a una distancia de 5 metros dado que después de 10 metros la Raspberry Pi se desconecta por lo cual se seleccionó un punto medio. Como tal para realizar las pruebas se tomó mediciones de tiempos apoyado en la librería time de python. Por medio de la consola de la Raspberry Pi se adquirieron esos tiempos para la conexión al servidor, almacenamiento de un solo dato y el tiempo de entrenamiento que toma la red neuronal implementada en la Raspberry Pi.

Los tiempos de procesamiento y almacenamiento se muestran en la tabla 1. Para la medición se realizaron 30 pruebas con muestras de dos pacientes diferentes de las cuales cada uno corresponden 192 vectores de 150 muestras, para las pruebas se usó un vector que representa una señal EEG seleccionada de manera aleatoria.

Para el entrenamiento de la red se usaron 112 vectores para entrenamiento y los 80 vectores restantes para pruebas de precisión.

Tabla 1. Tiempos promedio de procesamiento y almacenamiento

Etapa	Tiempo transcurrido (ms)
Entrenamiento de la red CNN	120,000 (2 minutos)
Acceso al servidor	600
Almacenamiento por muestra	9.6

De la tabla 1 se tienen tres aspectos. El primero de ellos es el tiempo transcurrido para entrenar la red CNN. De las 30 pruebas realizadas el tiempo de entrenamiento permanece constante en 2 minutos. No obstante, esta etapa de entrenamiento solo se realiza una vez por paciente. A su vez, el entrenamiento de la red generó un desempeño promedio de 93.75% en la correcta detección de la onda P300.

El segundo aspecto es el acceso al servidor, es decir, el tiempo que se toma para que un dispositivo pueda acceder a la base de datos y salir con seguridad. El retardo se da porque el servidor

debe autorizar la entrada del dispositivo. De las 30 pruebas de acceso el tiempo de acceso desde una Raspberry Pi permanece casi constante en 0.6 segundos, solo con una desviación estándar de 0.03195 ms.

El tercer aspecto es el tiempo que toma al servidor para almacenar un dato. El tiempo dependió de la carga que tuviera la Raspberry Pi, sin embargo no hubo grandes fluctuaciones en los tiempos.

4 CONCLUSIONES

Para el presente trabajo, se presentó una red IoT bajo un esquema de computación en el borde donde en un servidor centralizado se almacenan los datos de BCIs embebidos que consisten en OpenBCI® que adquieren señales EEG para ser procesadas con una Raspberry Pi para el estudio de la onda P300. Una vez realizado el procesamiento, la Raspberry Pi procede a transmitir la detección del P300 y la misma señal EEG que da acceso a diversos dispositivos para poder almacenar la información. La red es escalable, es decir, se pueden tener varias BCI embebidos y servicios de análisis conectados al servidor central. Cabe mencionar que no existe acceso a la nube.

De acuerdo a los resultados, se puede ver que el mayor retardo fue el de los tiempos de entrenamiento de 2 minutos. Esto se debe a la gran cantidad de muestras utilizadas y la capacidad de cómputo de la Raspberry Pi®. Sin embargo, este retardo solo se tiene una vez por cada paciente ya que la red CNN solo necesita entrenarse una vez para ajustar el modelo del paciente. El acceso al servidor demanda cierto tiempo debido al hardware utilizado y la velocidad de la red que se tiene, una mejora en estos dos factores puede resultar en un menor tiempo de acceso. Almacenar las señales una no denota un lapso de tiempo significativo lo cual muestra un buen desempeño en este aspecto.

En comparación con los BCI conectados a Internet, estos tiempos son muy reducidos, ya que no existe la latencia derivada de la distancia, permisos y retardos de los servidores de la nube. Sin embargo, un problema de la red propuesta es que el adaptador de la Raspberry Pi® no permite manejar grandes distancias y el router limita la cantidad de direcciones IP a 100.

Como trabajos a futuro el servidor central puede mejorar su velocidad de procesamiento conectando la red por medio de Ethernet al router y esta sería la que maneja la conexión inalámbrica del sistema. Otro punto interesante que no se probó en este trabajo es anexar más servicios de aplicación dentro de la etapa de servicio de análisis.

AGRADECIMIENTOS

Los autores agradecen a Tecnológico Nacional de México por el apoyo TecNM 6418.18-P para la realización de este trabajo.

5 REFERENCIAS.

- [1] T. W. a. B. H. Anirudh Vallabhaneni, «Brain-computer interface,» *ResearchGate*, p. 35, 2015.
- [2] B. He, B. Baxter, B. J. Edelman, C. C. Cline y W. W. Ye, «Noninvasive Brain-Computer Interfaces Based on Sensorimotor Rhythms,» *Proceedings of the IEEE*, vol. 103, n° 6, 2015.
- [3] J. R. Wolpaw y D. J. McFarlan, «Control of a two-dimensional movement signal by a,» *Proceedings of the National Academy of Sciences*, vol. 101, n° 51, 2004.
- [4] Fabio Babiloni, D. M. Fabio Aloise, S. Bufalari y Andrea Tocci, «Non Invasive EEG-Based Brain Computer Interface for Communication and Control,» *International Journal of Bioelectromagnetism*, vol. 7, n° 1, p. 2, 2007.
- [5] F. Galan y L. E. F. P. V. G. P. J. d. R. M. J. Nuttin M., «A brain-actuated wheelchair: Asynchronous and non-invasive,» *Clinical Neurophysiology Elsevier*, vol. 119, 2008.
- [6] J. Meng, S. Zhang, A. Bekyio, J. Olsoe, B. Baxter y B. He, «Noninvasive Electroencephalogram Based Control of a Robotic Arm for Reach and Grasp Tasks,» *scientific reports*, 14 December 2016.
- [7] S. P., «A State of the Art review on the internet of things,» de *Conferencia de IEEE*, Chennai India, 2014.
- [8] X. Zhang, L. Yao, S. Zhang, S. Kanhere, Q. Sheng y y. Liu, «Internet of Things Meets Brain-Computer Interface: A Unified Deep Learning Framework for Enabling Human-Thing Cognitive Interactivity,» *Journal of Latex Class files*, vol. 14, n° 8, 2015.
- [9] D. Evans, «Internet de las cosas: como la proxima evolucion de internet cambia todo,» cisco, San jose CA, 2011.
- [10] M. Weiser, «Ubiquitous Computing,» *IEEE Computer "Hot Topics"*, vol. 26, n° 10, pp. 71-72, 1993.
- [11] A. Soares Teles, M. Cagy, F. Silva, M. Endler, V. Bastos y S. Teixeira, «Using Brain-Computer Interface and Internet of Things to Improve Healthcare for Wheelchair Users,» de *UBICOMM*, 2017.
- [12] C. G. Coogan y B. He, «Brain-Computer Interface Control in a Virtual Reality Environment and Applications for the Internet of Things,» *IEEE Access*, vol. 6, 2018.
- [13] M.-P. Hosseini, D. Pompili, K. Elisevich y H. Soltanian-Zadeh, «Optimized Deep Learning for EEG Big Data and Seizure Prediction BCI via Internet of Things,» *IEEE TRANSACTIONS ON BIG DATA*, vol. 3, n° 4, 2017.
- [14] S. K. R., «A study of cloud computing environments for Hifh Performance applications,» de *2016 International Conference of Data Mining and Advanced Computing (SAPIENCE)*, Ernakulam, India, 2016.
- [15] J. C. Q. Z. Y. L. L. X. Weisong Shi, «Edge Computing: Vision and Challenges,» *IEEE INTERNET OF THINGS JOURNAL*, vol. 3, n° 5, p. 36, 2016.
- [16] J. M. Macías Macías, «Desarrollo e implementación de algoritmo clasificador de señal P300 en un sistema embebido,» Instituto Tecnológico de Chihuahua, Chihuahua, Chihuahua, Mexico, 2017.
- [17] R. Cattell, «Scalable SQL and NoSQL data stores,» *ACM SIGMOD Record*, vol. 39, n° 4, pp. 12-27, 2010.