UNIVERSIDAD DE SALAMANCA

Departamento de Informática y Automática

TESIS DOCTORAL

Metodología multi-criterio de optimización de recursos en sistemas embebidos para implementación de algoritmos de clasificación supervisados

Autor:
Paul David Rosero Montalvo

2020

UNIVERSIDAD DE SALAMANCA

Departamento de Informática y Automática
Facultad de Ciencias
Grupo de investigación:
MIDAS

AUTOR:

Paul David Rosero Montalvo

DIRECTOR:

Vivian Félix López Batista, PhD.

Declaración de Autoría

La Dra. Vivian Félix López Batista, Profesora Titular de Universidad del Departamento de Informática y Automática de la Universidad de Salamanca.

HACE CONSTAR:

que el doctorando Paul Rosero Montalvo ha desarrollado el trabajo titulado "Metodología multicriterio de optimización de recursos en sistemas embebidos para implementación de algoritmos de clasificación supervisados" bajo su supervisión, y por ello autoriza su presentación para la obtención del título de Doctor.

DIRECTOR:

LOPEZ BATISTA
VIVIAN FELIX -
12420865Z

Firmado digitalmente por
LOPEZ BATISTA VIVIAN
FELIX - 12420865Z
Fecha: 2020.10.16 10:05:19
+02'00'

Vivian Félix López Batista, PhD.

Salamanca, 16 de octubre del 2020

DEDICATORIA

Esto es para los locos, para los inadaptados, los rebeldes, los alborotadores, aquellos que no encajan, aquellos que ven las cosas de manera diferente, para aquellos que no les gustan las reglas y no respetan el status quo. Los puedes citar, puedes estar de acuerdo con ellos, puedes glorificarlos o insultarlos, pero lo único que seguro no puedes hacer es ignorarlos.
Porque son los que cambian las cosas, son los que impulsan a la raza humana hacia adelante.

Mientras algunos los ven como unos locos, nosotros vemos su genialidad. Porque las personas que están tan locas para pensar que pueden cambiar el mundo, son aquellas que lo hacen y lo logran.

Steve Jobs.

AGRADECIMIENTOS

Este trabajo fue realizado gracias el apoyo de varias personas que quisiera tomarme el tiempo para cada una de ellas:

A mi tutora Vivian López, por su apoyo para la presentación de esta tesis que con sus recomendaciones en docencia y su experiencia en el campo de la investigación, han sido clave para mi formación como un investigador. Esto deriva a agredecer a la Universidad de Salamanca, que me recibió como un estudiante de doctorado y que con su profesorado, personal adminsitrativo e instalaciones, permitieron la consecución de este trabajo.

A mi gran amigo Diego Peluffo, por el acompañamiento y asesoría en todos estos artículos de alto impacto que hemos trabajado juntos. Además, fue quién me sugirió y motivó a seguir los estudios de doctorado. En este sentido, al grupo de investigación SDAS-GROUP que él lidera, por la colaboración en el desarrollo de los sistemas electrónicos presentados en esta tesis.

A mi familia, especialmente a mis padres por contar con todo su apoyo para emprender en el camino de la investigación y haberme formado como una persona de bien, que no debe rendirse a pesar de circunstancias adversas.

A mis colegas investigadores por sus recomendaciones y trabajo en conjunto.

Paul

RESUMEN

En la actualidad, hemos visto un aumento en el uso de los sistemas embebidos, debido a su flexibilidad de instalación y su capacidad de recopilar datos por medio de sensores. Estos sistemas tienen como base la combinación entre las Tecnologías de la Información y la Comunicación (TIC), el concepto de Internet of Things (IoT) y la Inteligencia Artificial (IA). Sin embargo, muchos desarrolladores e investigadores, no realizan un proceso exhaustivo sobre la veracidad de la información que busca representar el fenómeno estudiado. Se debe tener en cuenta, que los valores obtenidos por los sensores, son una aproximación del valor real, debido a la transformación de la señal de naturaleza física hacia una eléctrica.

Esto ha ocasionado que la forma de almacenar dicha información esté más orientada a la cantidad que a la calidad. En consecuencia, la búsqueda de conocimiento útil a través de los sistemas embebidos, por medio de algoritmos de aprendizaje automático, se vuelve una tarea complicada. Tomando también en consideración, que el desarrollador del dispositivo electrónico, en ocasiones, no tiene un pleno conocimiento sobre el área de estudio donde va a ser empleado el sistema.

La presente tesis doctoral, propone una metodología multi-criterio de optimización de recursos en sistemas embebidos para la implementación de algoritmos de clasificación empleando criterios de aprendizaje automático. Para hacer esto, se busca reducir el ruido obtenido por el porcentaje de incertidumbre ocasionado por los sensores, mediante el análisis de criterios de acondicionamiento de la señal. Además, se ha visto que, emplear un servidor externo para el almacenamiento de datos y posterior análisis de la información, influye en el tiempo de respuesta del sistema. Por esta razón, una vez cumplida la tarea de encontrar una señal depurada, se realiza un análisis de los diferentes criterios de selección de características de los datos, que permitan reducir el conjunto almacenado, para cumplir dos funciones principales. La primera, evitar la saturación de servicios computacionales con información almacenada innecesariamente. La segunda, implementar estos criterios de aprendizaje automático, dentro de los propios sistemas embebidos, con el fin de que puedan tomar sus propias decisiones sin la interacción con el ser humano.

Esta transformación, hace que el sistema se vuelva inteligente, ya que puede elegir información relevante y cómo puede adaptarse a su entorno de trabajo. Sin embargo, la codificación de estos modelos matemáticos que representan los algoritmos de aprendizaje automático, deben cumplir requisitos de funcionalidad, basados en la capacidad computacional disponible en un sistema embebido. Por esta razón, se presenta una nueva clasificación de sistemas embebidos, con una novedosa taxonomía de sensores, enfocados a la adquisición y análisis de datos. Concretamente, se diseña un esquema de acoplamiento de datos entre el sensor y el sistema procesador de información, que brinda una recomendación de uso del criterio de filtrado de datos, en relación con la capacidad de recursos computacionales y la forma de envío de información dentro del sistema embebido. Este proceso se valida mediante métricas de rendimiento de sensores.

Por otra parte, una vez que se tenga una base de datos adecuada, se presenta una técnica de selección de los algoritmos basados en aprendizaje supervisado, que se ajuste a los requisitos de funcionalidad del sistema embebido y a su capacidad de procesar información. Específicamente, se analizan los criterios de selección de características, prototipos y reducción de dimensionalidad que se adapten a los diferentes algoritmos de clasificación para la elección de los más adecuados.

ABSTRACT

Nowadays, exist an increase in the use of embedded systems, due to their installation flexibility and their ability to collect data through sensors. These systems are based on the combination of Information and Communication Technologies (ICTs), the concept of Internet of Things (IoT) and Artificial Intelligence (AI). However, many developers and researchers do not make an exhaustive process on the veracity of the information that seeks to represent the studied phenomenon. It must be taken into account that the obtained values by sensors are an approximation of the real value, due to the transformation of the physical nature signal to an electrical one.

For these reasons, the way of storing this information is more oriented to quantity than to quality. Consequently, the search for useful knowledge through embedded systems, through machine learning algorithms, becomes a complicated task. Taking into consideration as well that the developer of the electronic device, sometimes, does not have full knowledge of the study area where the system will be used.

This doctoral thesis proposes a multi-criterion for optimizing resources methodology in embedded systems for the implementation of classification algorithms using machine learning criteria. To do this, it seeks to reduce the obtained noise caused by percentage uncertainty by the sensors, through the analysis of signal conditioning criteria. In addition, it has been seen that using an external server for data storage and subsequent analysis of the information influences the response time of the system. For this reason, once the task of finding a refined signal has been completed, an analysis of the different criteria for selecting the characteristics of the data is carried out, which allows reducing the stored set, to fulfill two main functions. The first is to avoid the saturation of computer services with unnecessarily stored information. The second one is to implement these machine learning criteria, within the embedded systems themselves, so that the system can make their own decisions without interaction with the human being.

This transformation makes the system become intelligent since it can choose the information that is relevant and how it can adapt to the work environment. However, the coding of these mathematical models that represent machine learning algorithms must meet functionality requirements, based on the computational capacity available in an embedded system. For this reason, a new embedded systems classification is presented, with a novel taxonomy of sensors, focused on the acquisition and analysis of data. Specifically, a data coupling scheme is designed between the sensor and the information processing system, which provides a recommendation for the use of the data filtering criteria, in relation to the capacity of computational resources and the way of sending information within the embedded system. This process is validated using sensor performance metrics.

Moreover, once an adequate database is available, a selection technique based on supervised learning algorithms is presented, which adjusts to the functionality requirements of the embedded system and its capacity to process information. Specifically, the characteristics selection criteria, prototypes, and dimensionality reduction that adapt to the different classification algorithms are analyzed to choose the most appropriate ones.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE CUADROS

Parte I

MODALIDAD DE LA TESIS

CAPÍTULO 1: MODALIDAD DE LA TESIS

La presente tesis doctoral de la Universidad de Salamanca, es realizada bajo el formato de compendio de artículos publicados previamente. Esta tesis incluye cuatro contribuciones publicadas en revistas y dos capítulos de libro.

1.1. Autorización de supervisión

Vivian Félix López Batista, Profesora Titular del Departamento de Informática y Automática de la Universidad de Salamanca y directora de la tesis doctoral de Paul David Rosero Montalvo.

Autoriza:

Que, Paul Rosero presente y defienda su tesis doctoral en la modalidad de compendio de artículos.

LOPEZ
BATISTA
VIVIAN FELIX -
12420865Z

Firmado digitalmente
por LOPEZ BATISTA
VIVIAN FELIX -
12420865Z
Fecha: 2020.10.16
10:06:43 +02'00'

Vivan Félix López Batista

1.2. Lista de contribuciones

1.2.1. Contribución 1

Título: Intelligent System for Identification of Wheelchair User's Posture Using Machine Learning Techniques, in IEEE Sensors Journal, vol. 19, no. 5, pp. 1936-1942, 1 March, 2019

DOI: 10,1109/$JSEN$,2018,2885323

Disponible en: https://ieeexplore.ieee.org/abstract/document/8565996

Autores:

- Paul D. Rosero-Montalvo
- Diego H. Peluffo-Ordóñez
- Vivian F. López-Batista
- Jorge Serrano
- Edwin A. Rosero-Rosero.

Revista: IEEE Sensors

Índices de Calidad:

- ***WoS JCR Impact Factor 2019***: 3.073. *Rank:* 91/266 (Q2). *Area:* Engineering, Electrical and Electronic.
- ***Índice Scopus***: 6.2. *Rank:* 9/129 (Q1). *Area:* Instrumentation.

1.2.2. Contribución 2

Título: Intelligent WSN System for Water Quality Analysis Using Machine Learning Algorithms: A Case Study (Tahuando River from Ecuador). Remote Sensing. 12(12): 1988 (2020)

DOI: *https* : //*doi.org*/10,3390/rs12121988

Disponible en: https://www.mdpi.com/2072-4292/12/12/1988

Autores:

- Paul D. Rosero-Montalvo
- Vivian F. López-Batista
- Jaime A. Riascos
- Diego H. Peluffo-Ordóñez

Revista: Remote Sensing

Índices de Calidad:

- ***WoS JCR Impact Factor 2019***: 4.509. *Rank:* 9/30 (Q2). *Area:* Remote Sensing.
- ***Índice Scopus***: 6.1. *Rank:* 15/187 (Q1). *Area:* Science.

1.2.3. Contribución 3

Título: Environment Monitoring of Rose Crops Greenhouse Based on Autonomous Vehicles with a WSN and Data Analysis.

Aceptada su publicación

Autores:

- Paul D. Rosero-Montalvo
- Vanessa Erazo-Chamorro
- Vivian F. López-Batista
- María Moreno-García
- Diego H. Peluffo-Ordóñez

Revista: MDPI Sensors

Índices de Calidad:

- ***WoS JCR Impact Factor 2019***: 3.275. *Rank:* 15/64 (Q1). *Area:* Instruments-Instrumentation.
- ***Índice Scopus***: 5.0. *Rank:* 17/129 (Q1). *Area:* Engineering.

1.2.4. Contribución 4

Título: Air Pollution Monitoring Using WSN nodes with Machine Learning Techniques: A case study.

Aceptada su publicación

Autores:

- Paul D. Rosero-Montalvo
- Vivian F. López-Batista
- Ricardo Arciniega-Rocha
- Diego H. Peluffo-Ordóñez

Revista: Logic Journal of the IGLP

Índices de Calidad:

- ***WoS JCR Impact Factor 2019***: 0.93. *Rank* 126/324 (Q3). *Area:* Mathematics. *Rank* 3/21 (Q1). *Area:* Logic.
- ***Índice Scopus***: 1.6 *Rank:* 70/606 (Q1). *Area:* Philosophy.

1.2.5. Contribución 5

Título: Multivariate Approach to Alcohol Detection in Drivers by Sensors and Artificial Vision. In: Ferrández Vicente J., Álvarez-Sánchez J., de la Paz López F., Toledo Moreo J., Adeli H. (eds) From Bioinspired Systems and Biomedical Applications to Machine Learning. IWINAC 2019. Lecture Notes in Computer Science, vol 11487. Springer, Cham.

DOI: $https : //doi.org/10{,}1007/978 - 3 - 030 - 19651 - 6\_23$

Disponible en: https://link.springer.com/chapter/10.1007/978-3-030-19651-6_23

Autores:

- Paul D. Rosero-Montalvo
- Vivian F. López-Batista
- Ricardo Arciniega-Rocha
- Vanessa Erazo-Chamorro
- Diego H. Peluffo-Ordóñez

Libro: Lecture Notes in Computer Science

Serie: *International Work-Conference on the Interplay Between Natural and Artificial Computation, vol 11487*

Índices de Calidad:

- ***Índice Scopus***: 1.9 *Rank:* 95/221 (Q3). *Area:* Computer science.

1.2.6. Contribución 6

Título: Urban Pollution Environmental Monitoring System Using IoT Devices and Data Visualization: A Case Study. In: Pérez García H., Sánchez González L., Castejón Limas M., Quintián Pardo H., Corchado Rodríguez E. (eds) Hybrid Artificial Intelligent Systems. HAIS 2019. Lecture Notes in Computer Science, vol 11734. Springer, Cham.

DOI: $https : //link.springer.com/chapter/10{,}1007/978 - 3 - 030 - 29859 - 3\_58$

Disponible en: https://link.springer.com/chapter/10.1007/978-3-030-29859-3_58

Autores:

- Paul D. Rosero-Montalvo
- Vivian F. López-Batista
- Leandro Lorente-Leyva
- Xiomara Blanco-Valencia
- Diego H. Peluffo-Ordóñez

Libro: Lecture Notes in Computer Science

Serie: *International Conference on Hybrid Artificial Intelligence Systems, vol 11734*

Índices de Calidad:

- ***Índice Scopus***: 1.9 *Rank:* 95/221 (Q3). *Area:* Computer science..
- ***Computing Research and Education (CORE)***: *Ranking:* C. *Area:* Computing Research

Parte II

PRELIMINARES

CAPÍTULO 2: INTRODUCCIÓN

El presente capítulo, se describe el área del conocimiento específico en el que se encuentra enfocada esta tesis doctoral. Por su parte, la sección 2.1 muestra la introducción. Posteriormente, se presenta la motivación en la sección 2.2 y la metodología de investigación en la sección 2.3. La hipótesis de partida y los objetivos se muestran en la sección 2.4 y 2.5 respectivamente. Finalmente, la presentación de la estructura de la memoria en la sección 2.6.

2.1. Introducción

Un sistema embebido se define como un conjunto de elementos electrónicos que conforman una unidad de procesamiento, bloques de memoria, bloques de entrada y salida de datos, un medio de comunicación y una batería de alimentación [3]. Conviertiéndose en un elemento fundamental del Internet de las Cosas (IoT por sus siglas en inglés *Internet of Things*) por su creciente uso en diferentes aplicaciones en la industria, automatización, salud, edificios inteligentes, entre otros [4,5]. Esto se debe a su flexibilidad de instalación y a su capacidad de recopilación de grandes volúmenes de datos en tiempo real. Además, su objetivo principal es proporcionar información para la toma de decisiones [6,7]. Todo esto, bajo diferentes protocolos de comunicación cableados o inalámbricos acorde a los requisitos de la apliación a ser empleados.

Este proceso de recolección de datos, es realizado por un elemento electrónico denominado transductor, este transforma una cantidad física en una eléctrica [8]. Además, se necesita emplear un circuito de acoplamiento de la señal para contar con una señal medible y estable. Como resultado, la unión del transductor y su circuito de acoplamiento, se denomina sensor [9, 10]. No obstante, durante el proceso de adquisición, los datos pueden verse influenciados por variables no controladas, así como factores ambientales que perjudican su validez y usabilidad. Considerando que los componentes físicos que integran a los sensores, están lejos de ser ideales e introducen componentes en voltaje continuo. Esto puede ocasionar incertidumbre en el proceso de medición y tratamiento de la señal. En consecuencia, sólo se puede representar una estimación del fenómeno físico a estudiar [11,12].

Una de las características principales de un sistema embebido es su capacidad de adaptación. Es decir, que pueda emular algunas habilidades de procesamiento que realiza el cerebro humano, lo que lo convierte en un sistema inteligente. Esto implica que de alguna manera, cuenta con la capacidad de tomar decisiones, aprender de estímulos externos y adaptarse a los cambios de su entorno mediante la posibilidad de ejecutar algoritmos matemáticos complejos [13]. Implícitamente, se basa en un paradigma computacional que recibe o procesa datos para lograr una tarea encomendada [8]. Bajo este concepto, se espera que los sensores puedan brindar la mejor información posible. En este sentido, una de las formas de tomar decisiones con parámetros establecidos, son los algoritmos de aprendizaje supervisados. Los cuales necesitan, por un lado, un conjunto de datos para el entrenamiento del modelo y por el otro, un conjunto de validación. Como se comentó con anterioridad, los datos recolectados por los sensores pueden tener muchos errores de lectura o información irrelevante. Por estas razones, no es recomendable almacenarlos todos [14–16].

Una de las formas para reducir el tamaño de las bases de datos, es a través de las técnicas de eliminación de redundacia o ruido. Éstas surgen como alternativas para minimizar el impacto de grandes conjuntos de datos en el comportamiento de los algoritmos, reduciendo su tamaño sin afectar la calidad del conocimiento intrínseco almacenado inicialmente [17, 18]. Es decir, los requisitos de manejo de datos se reducen mientras que la capacidad predictiva de los algoritmos se mantiene. Consecuentemente, hay que considerar las limitaciones de recursos computacionales de un sistema embebido [15, 19]. Teniendo en cuenta que mientras más datos sean procesados, el tiempo de vida de la batería y de toma de decisión, se verán reducidos proporcionalmente [20].

En esta tesis doctoral, se presenta la integración de dos áreas: El diseño, desarrollo y optimización de un sistema embebido y el análisis de datos correspondiente con los algoritmos de aprendizaje automático, con el fin de ser compilados en estos sistemas. Para hacer esto, se propone una nueva metodología que sea capaz de determinar la mejor técnica de adquisición y selección de los datos que no interfiera en el rendimiento del clasificador [21]. Además, los algoritmos de aprendizaje automático puedan ser procesados en un sistema embebido de bajo recursos computacionales. De esta forma, obtener una propuesta que ayude al desarrollo de aplicaciones IoT con datos multidimensionales [22].

2.2. Motivación

Actualmente, se cuenta con sensores muy precisos de bajo coste que interactúan con microcontroladores de reducidas capacidades computacionales. Estos son muy utilizados por su flexibilidad de implementación en diferentes aplicaciones electrónicas. No obstante, existe el reto de adquirir información veraz sobre el fenómeno estudiado, ya que en muchos casos, no se toman consideraciones de validar el proceso de adquisición de datos de estos sistemas con respecto a instrumentos robustos que permitan cerciorarse de la calidad de los datos. Además, el desarrollo del sistema electrónico, debe contar con un proceso estadarizado que conlleve una adecuada selección de elementos y que derive en un óptimo uso de recursos computacionales. A su vez, el proceso de acoplamiento y adquisición de datos, debe estar acorde con la aplicación y el tipo de sensor a ser empleado. Por esta razón, se debe contar con un esquema de desarrollo de sistemas embebidos que se encuentre enfocado a una adaptación desde el mundo real a un entorno digital. Además, que permita seleccionar los diferentes elementos electrónicos bajo recomendaciones técnicas orientadas al análisis de datos. Con ello, el sistema embebido tiene la posibilidad de incorporar la capacidad de toma de decisiones. Esto evita que el incremento de dispositivos conectados a internet que almacenan y procesan datos en la nube, realicen el análisis de datos en servidores externos, teniendo en consideración que existen ciertas aplicaciones de IoT que necesitan un procesamiento en tiempo real. En consecuencia, su respuesta puede verse influenciada por la latencia y disponibilidad de la red de acceso.

Por esta razón, es necesario desarrollar sistemas que puedan procesar los datos de forma local. Proporcionando la capacidad de reacción inmediata en una capa de estandarización que facilite la comunicación con la nube. Como resultado, solo se enviará la información necesaria a internet. Este concepto se conoce como la niebla de IoT o *Edge Computing* [23].

2.3. Metodología de investigación

La metodología empleada permite identificar y formular el problema a partir de una hipótesis que ha sido fundamentada en un análisis sistemático de literatura que permite establecer una propuesta de solución. Con ello, se definen los objetivos de investigación que derivan en las actividades a realizar y que su consecución dará paso a la formulación de las conclusiones de la presente tesis doctoral. Por esta razón, se establecen las etapas siguientes:

- **Definición de la problemática:** Planteamiento del problema que permite establecer los objetivos y la hipótesis de la investigación.
- **Revisión de estado del arte:** Análisis de las soluciones planteadas hacia la problemática presentada tanto en técnicas como en métodos llevados a cabo. Este proceso es constante a lo largo de la investigación.
- **Acoplamiento y adquisición de datos:** Desarrollo de sistemas electrónicos multivariantes con diferentes sensores, aplicando criterios de adquisición y acondicionamiento de señales que derive en un adecuado proceso de recopilación y almacenamiento de datos.
- **Evaluación de casos de estudio:** Desarrollo de un enfoque multi-criterio para la implementación de algoritmos de aprendizaje automático dentro de un sistema embebido, integrando fases de acoplamiento de señal con criterios de *software*, *hardware* y analisis de datos de bajo coste computacional.
- **Estudio de algoritmos supervisados:** Estudio comparativo de algoritmos de clasificación con diferentes métricas de rendimiento aplicado a bases de datos provenientes de sistemas embebidos que previamente han sido pre-procesadas con técnicas de selección de prototipos, selección de variables y reducción de dimensionalidad.
- **Publicación de los resultados:** Presentación de resultados parciales y finales tanto en conferencias como en revistas de alto impacto que promuevan el intercambio de ideas y la validación de la propuesta de tesis doctoral.

2.4. Hipótesis

La hipótesis de esta tesis doctoral recae en que la conjugación de sistemas embebidos con técnicas de análisis de datos *in-situ* puede alcanzar un equilibrio entre el coste computacional y la precisión a través de una metodología multicriterio que optimice simultáneamente el consumo de recursos del sistema embebido y el desempeño del análisis de datos. Este proceso, se debe iniciar desde un adecuado esquema de acoplamiento de datos entre los sensores empleados y el microcontrolador que conforman el sistema embebido. Posteriormente, una evaluación de los diferentes criterios de limpieza y reducción de datos, se debe llevar a cabo para elegir al algoritmo adecuado que se adapte a la naturaleza de la información obtenida. Finalmente, con la base de datos depurada, se determina el correspondiente algoritmo de clasificación supervisado que tenga el mejor rendimiento entre el menor número de instancias empleadas para el entrenamiento del modelo y un alto acierto de clasificación. Este contexto ha sido muy poco explorado y existen diversos problemas tales como:

- Las actuales clasificaciones y taxonomías en sistemas embebidos y sensores no se relacionan con las nuevas tendencias sobre el uso de algoritmos de aprendizaje automático. Por esta razón, las etapas de filtrado y suavizado de la señal son enfocadas de forma específica a la tarea a realizar, sin contar por el momento con un esquema de adquisición de datos de propósito general.

- Existen varios criterios sobre los procesos de limpieza y reducción de datos por medio de los algoritmos de aprendizaje automático. No obstante, éstos son diseñados para ser compilados en servidores y no hacia una creciente demanda de los sistemas embebidos.
- Por su parte, los algoritmos de clasificación supervisada tienden a emplear un conjunto de entrenamiento con el mayor volumen de datos posible. Sin embargo, esto puede ocasionar un sobre ajuste del modelo, mayor tendencia de asignación a una etiqueta, entre otros. Esto deriva en un mayor tiempo de respuesta del sistema, ya que necesita aumentar el consumo de los recursos computacionales. Con ello, la tarea de implementar estos algoritmos en sistemas embebidos, debe ser llevada a cabo con una etapa previa de reducción del conjunto de datos de entrenamiento para la generación del modelo de clasificación.

La hipótesis de esta tesis doctoral, pretende validar que la presente metodología de optimización multi-criterio permite contar con un adecuado esquema de adquisición y análisis de datos en sistemas embebidos que proporciona la apropiada información en la representación de un fenómeno estudiado.

Para verificar dicha metodología, se diseñaron diferentes casos de estudio en varios campos del conocimiento donde los sistemas embebidos puedan tomar datos, procesarlos y realizar una clasificación de estados en base a su funcionalidad. Todo este proceso es valorado con diferentes métricas de rendimiento, empezando desde la calibración y validez de la información por parte de los sensores hasta el porcentaje de acierto de los algoritmos de clasificación.

2.5. Objetivos

Para poder validar esta hipótesis, es necesario establecer un enfoque que permita abordar y afrontar la problemática que supone. Por ello, el objetivo principal de este trabajo es:

2.5.1. Objetivo general

Desarrollar una metodología de optimización de recursos multicriterio en sistemas embebidos orientado a la implementación y evaluación de algoritmos de clasificación supervisados y representación de datos multivariantes

2.5.2. Objetivos específicos

- Proponer un esquema de adquisición y acondicionamiento de datos multivariantes provenientes de sensores en sistemas embebidos que alcancen un buen compromiso entre métricas de precisión y exactitud.
- Seleccionar una técnica de representación de datos a partir de un estudio comparativo de los diferentes enfoques de análisis de datos, con el fin de determinar el criterio adecuado que permita obtener un conjunto óptimo de entrenamiento.
- Diseñar un enfoque multicriterio para optimizar el coste computacional de los algoritmos de clasificación supervisados que sea acorde con la representación del conjunto de entrenamiento por medio de métricas de rendimiento.

2.6. Estructura de la memoria

La presente tesis doctoral se encuentran en la modalidad de compendio por artículos científicos. El capítulo II, corresponde a la presentación de la modalidad de la tesis. Por su parte, el capítulo III, muestra el contexto y el estado del arte correspondiente a la temática de la tesis doctoral. La sección IV, presenta la coherencia y relación directa entre los artículos/publicaciones presentados y los objetivos de la tesis doctoral. Donde se muestran los objetivos, metodología, resultados y conclusiones de cada publicación. Finalmente, se presentan los anexos de cada contribución en el capítulo V.

Parte III

CONTEXTO Y ESTADO DEL ARTE

CAPÍTULO 3: CONTEXTO Y ESTADO DEL ARTE

El presente capítulo, muestra de forma teórica las bases necesarias para el desarrollo de esta tesis doctoral. Empezando desde el estudio de los trabajos relacionados y las diferentes propuestas planteadas con respecto al proceso de adquisición y el análisis de datos en sistemas embebidos. Por esta razón, se encuentra dividido en dos partes. La primera, se enfoca en el diseño electrónico y sus componentes relacionados con los sistemas embebidos y sensores (secciones 3.1, 3.2 y 3.3). En la segunda, se muestra el aprendizaje supervisado y las diferentes técnicas y algoritmos relacionados con la temática planteada (sección 3.4).

El desarrollo de sistemas embebidos se ha convertido un campo de investigación importante. Esto se debe a su capacidad de recopilar datos de diferentes áreas del conocimiento. En relación a las condiciones que los sistemas embebidos funcionan, existen diferentes formas de procesar y adquirir datos provenientes principalmente de los sensores. Por esta razón, al realizar un análisis de búsqueda sistematizada de información, se logra determinar cronológicamente los trabajos más relevantes relacionados al desarrollo de sistemas embebidos. Se empieza desde el desarrollo de sistemas embebidos enfocados en la recolección de datos (años 2009) hasta la actualidad.

Este estudio inicia con [24], que presenta un sistema embebido basado en sensores de gases para monitorear y clasificar una gran cantidad de olores industriales. Para ello, hace uso de una red neuro-difusa y un envío de datos por medio del protocolo inalámbrico *ZigBee*. En años posteriores (2010-2012), los sistemas embebidos se enfocan en la detección de humo para la prevención de incendios al usar ciertos criterios de redes neuronales que se compilan en un servidor externo [25]. No obstante, en el año 2013, existe un auge tecnológico en la portabilidad en sistemas de adquisición de bioseñales. Como resultado, se emplea como novedad, el concepto de textil inteligente [26]. Otro aspecto muy relevante en el mismo año, es el uso de cámaras embebidas para la clasificación de imágenes [27].

Con el amplio avance tecnológico (2014), trabajos como [28,29] implementan criterios de sistemas en tiempo real (RTC por su siglas en inglés *Real Time-Clock*) para la optimización de recursos computacionales, permitiendo alargar el tiempo de vida de las baterías. En estos trabajos, se desarrolla el reconocimiento de gestos y un estudio a fondo de una señal electromiográfica en sistemas embebidos. Posteriormente (2015-2016), los sistemas embebidos se vuelven más robustos debido al desarrollo de procesadores digitales de señales (DSP por sus siglas en inglés *Digital Signal Processing*) embebidos. Con ello, pueden procesar mayor cantidad de información y adquirir datos con mayor precisión. Como resultado, [30] presenta una solución de monitero de calidad del aire y [31] un análisis de datos para el envío de información a un servidor IoT que emplea algoritmos de aprendizaje supervisado. En el año siguiente (2017), toman mayor importancia el desarrollo de textiles inteligentes con sistemas RTC en enfoques médicos en el cuidado del adulto mayor. Un ejemplo de ello es [32]. Por otra parte, [33] realiza un sistema de detección de hongos en hogares con uso de cámaras.

En los años siguientes (2018-2020), empiezan por un lado, con el desarrollo de sistemas híbridos que integran sensores y visión artificial para diferentes tareas de clasificación como el reconocimiento de estrés [34] e interfaces persona-ordenador [35]. Por otro lado, existe un creciente interés sobre el monitoreo de condiciones ambientales en entornos urbanos empleando protocolos de comunicación inalámbrica de largo alcance como lo es *Low-Power Wide-Area Network* (LPWAN). Finalmente, trabajos como [36, 37] presentan soluciones de agrilcultura inteligente y monitoreo de bosques.

Con este análisis, se puede deducir que la implementación de algoritmos de aprendizaje automático en sistemas embebidos para la extración de conocimiento a partir de la adquisición de datos es viable. Siendo los algoritmos más utilizados los basados en aprendizaje supervisado, que utilizan criterios probabilísticos, los basados en distancias y los heurísticos. No obstante, en la actualidad existe una tendencia a aplicar el aprendizaje profundo o *deep learning* y los basados en modelos. Sin embargo, la mayor parte de estos trabajos, no se ha considerado la validación de los datos provenientes de los sensores y en consecuencia, los sistemas embebidos no pueden ser probados en entornos reales. Finalmente, estos sistemas desarrollados han significado un gran aporte para el aumento de repositorios digitales que recopilan información de los estudios realizados en diferentes áreas del conocimiento.

3.1. Sistemas embebidos

Un microprocesador es la unión de componentes electrónicos agrupados en un solo circuito integrado que funcionan como una unidad de procesamiento central o *Central Processor Unit* (CPU) [38]. La misma, mediante buses de comunicación interna puede realizar el intercambio de información entre los periféricos de entrada y salida. Además, es necesario el uso de diferentes tipos de memorias para el almacenamiento y procesamiento de datos. Como resultado, la unión de todos estos elementos, es considerado como un sistema microcontrolador [39].

Actualmente, en relación a la empresa que lo desarrolla y su campo de aplicación, pueden variar sus características internas para mejorar su desempeño. Por lo tanto, en términos generales y en relación a su capacidad computacional, cuenta internamente con conversores análogos-digitales y digitales-análogos, interrupciones de estados, contadores de tiempo, modos de ahorro de energía, diferentes tipos de registros, entre otros [40]. En la Fig. 3.1 se muestra un esquema general de un microncontrolador.

Este microcontrolador puede ejecutar código e interacturar con todos sus perifericos. Sin embargo, al contar con recursos computacionales limitados, su programación debe ser muy concisa en relación a su aplicación (propósito específico). En cuanto al desarrollo con estos sistemas electrónicos y con la tendencia en el uso de plataformas de *hardware* y *software* libre, como es el caso de `Arduino` y `Raspberry Pi`. Se han puesto a disposición de los usuarios sus diagramas eléctricos y especificaciones técnicas, con el objetivo de contar con una gran contribución para su perfecionamiento por medio de la comunidad de desarrolladores de aplicaciones electrónicas. Cabe destacar, que este avance tecnológico, ha permitido generar nuevas formas de diseño de microcontroladores con el objetivo de integrar varios de ellos para aumentar su capacidad de procesamiento [41]. Estos son denominados como sistemas en un chip (SoC por su siglas en inglés *System on a Chip*) [42]. Con ello, se pueden contar con capacidades de implementar algoritmos de aprendizaje automático

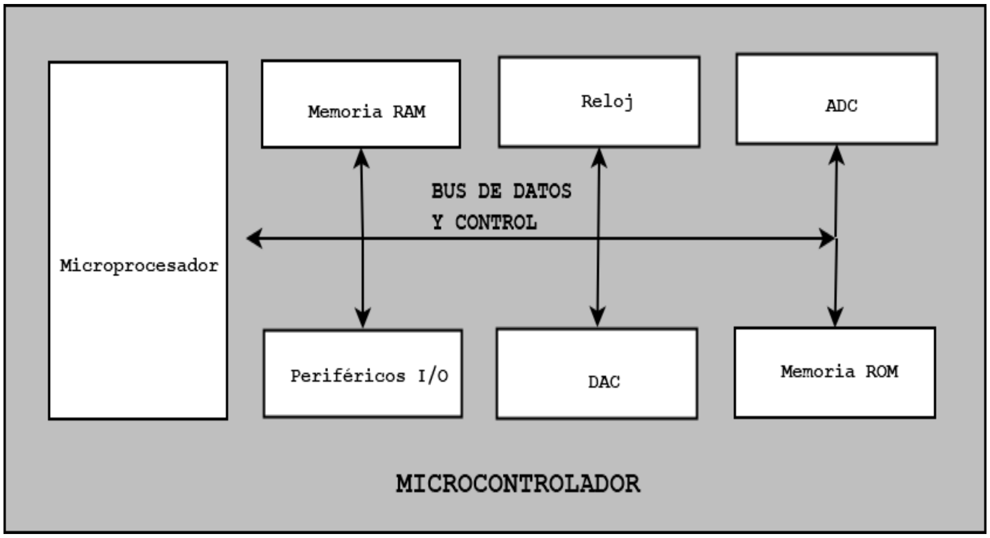


Figura 3.1: Diseño interno general de un microcontrolador

y toma de datos en tiempo real. No obstante, estos sistemas se encargan solo de aplicaciones específicas y están diseñados para cada una de ellas. Por ello, se ve la necesidad de integrar sensores, para recopilar datos de su entorno y un medio de comunicación que permita enviar información a un servidor o al IoT. Como resultado, este proceso permite la extracción de conocimiento intrínseco que existe en el conjunto de datos adquirido. Por estas razones, se han denominado como sistemas embebidos (SE) o sistemas empotrados [43]. Debido a su usabilidad, existen cuatro tipos principales: textiles inteligentes (sección 3.1.1), redes de sensores inalámbricas (sección 3.1.2), sistemas en tiempo real (sección 3.1.3), sistemas de aplicaciones específicas (sección 3.1.4) y ordenadores de placa reducida (sección 3.1.5). No obstante, para validar sus funcionalidades, es necesario realizar pruebas de *hardware* y *software* en estos sistemas para conocer su capacidad de interacción con el entorno (sección 3.1.6). Estos criterios son implementados en conjunto con algoritmos de aprendizaje automático, se convierte en un sistema inteligente. Este sistema puede actuar y guardar el resultado de las acciones tomadas para luego realizar una retroalimentación y aprender de la experiencia. Sus principales funcionalidades pueden ser: industria, sistematización, capacidad sensorial, reglas de actuación, transporte, entre otros [43].

3.1.1. Textiles inteligentes

Una de las principales precupaciones del ser humano en los últimos años es la detección temprana de enfermedades. Sin embargo, con el aumento de la población, se ha vuelto una tarea complicada de realizar. Sobre todo, en pacientes que puedan ser propensos a sufrir algunas enfermedades difíciles de detectar con revisiones médicas eventuales. Por esta razón, los sistemas embebidos en conjunto con sensores ubicados en el cuerpo, ayudan a revolucionar el campo médico al proporcionar una fuente de datos de pacientes, al recopilar información continuamente. Esta información, puede ser usada para desarrollar planes médicos que se enfocan en mejorar la salud (horas de sueño y ejercicio), detección temprana de enfermedades y proporcionar una alerta de eventos peligrosos, por ejemplo, caídas y ataques cardíacos. Hay que recalcar, que la cantidad de datos recopilados, incluso por un pequeño conjunto de sensores que funcionan todo el día es demasiado para que cualquier persona las analice. Por esta razón, el procesamiento y la clasificación de la señal lo debe realizar un sistema experto. Como resultado, se extrae automáticamente información útil de los pacientes [40, 44].

En relación con la recopilación de datos de señales del cuerpo (bio-señales), una de las soluciones planteadas es el uso de prendas textiles que en su interior contengan sistemas embebidos de monitoreo de señales o adquieran datos de estímulos externos como la detección en el aumento o disminución de temperatura ambiental o corporal, con el fin de brindar una alerta. Por lo tanto, si este dispositivo se encuentra en contacto con el ser humano, debe tener algunas características como: permeabilidad, flexibilidad, durabilidad, entre otros. Con ello, son capaces de alterar su naturaleza en respuesta a la acción de diferentes estímulos externos, físicos o químicos que modifican alguna de sus propiedades para generar un beneficio al ser humano. Como resultado, se han convertido en una de las herramientas más importantes para nuevas tecnologías como la telesalud. No obstante, su proceso de recopilación de datos es complicado debido al movimiento normal de las personas, el natural desgaste de sus elementos y los bajos niveles de las señales eléctricas del cuerpo humano que se pueden adquirir [45, 46].

3.1.2. Redes de sensores inalámbricas

Las redes de sensores inalámbricas (WSN por sus siglas en inglés *Wireless Sensor Networks*) se pueden definir como un sistema embebido compuesto por bloques de memoria, periféricos de entrada-salida, puertos de comunicación y una batería. Por esta razón, una WSN se utiliza principalmente para la adquisición de datos en entornos donde los sistemas tradicionales son ineficaces o de alto coste para su implementación. Además, una ventaja de una WSN, es la integración de varios sensores que trabajan en conjunto y permiten la conversión de señales físicas a eléctricas, que pueden enviarse a un destino remoto como un servidor. Este proceso lleva a cabo mediante el uso de protocolos y tecnologías inalámbricas [47, 48].

Actualmente, las WSN todavía tienen varios desafíos en *hardware* y *software* por superar. Estos se centran en la confidencialidad y fiabilidad de los datos, capacidad de almacenamiento y disponibilidad de acceso al medio de comunicación. Adicionalmente, desde el punto de vista del *hardware*, hay desafíos como coste eléctrico, tiempo de ejecución, tamaño del código, duración de la batería, y calibración del sensor [49]. Para definir si un sistema es eficiente, varios autores han propuesto un proceso de diseño para un sistema integrado: (i) una especificación funcional, (ii) un conjunto de propiedades que el diseño debe cumplir, (iii) definir los requisitos para la evaluación del diseño y (iv) definir el rendimiento del sistema en condiciones reales [48].

La topología de red para una WSN dependerá de la aplicación que se vaya a realizar, además de intentar obtener la mejor configuración de los componentes para un correcto funcionamiento de la red. Para lo cual, existen tres topologías con diferentes características en dependencia del tipo de envío de datos [47].

- **Estrella:** Topología donde cada nodo tiene una conexión directa con un nodo central considerado como puerta de enlace.
- **Tipo árbol:** Los nodos se conectan con un nodo tipo ruteador hasta llegar a conectarse al nodo central.
- **Tipo malla:** Los nodos se pueden conectar con múltiples nodos tratando de enviar los datos por el camino que encuentre la red con mayor confiabilidad.

3.1.3. Sistemas en tiempo real

Los sistemas en tiempo real, son aquellos que generalmente no se encuentran relacionados con el usuario, ya que su prioriodad se enfoca en los procesos. En este sentido, su funcionamiento se basa en entornos donde existen gran cantidad de tareas que deben ser atendidas en un tiempo determinado. Como resultado, los tiempos de respuesta se vuelven óptimos. Entre sus aplicaciones más relevantes se encuentran: control de trenes, telecomunicaciones, sistemas multimedia, control de edificios, entre otros [42].

3.1.4. Sistemas de aplicación específica

En algunos casos, los sistemas embebidos necesitan recursos específicos para aplicaciones que requieren diferentes características computacionales, especialmente desde la estructura de la CPU. Uno de estos casos, son los sistemas DSP. Estos cuentan con funciones relacionadas con procesos matemáticos complejos para el análisis, filtrado y compresión de la señal. Generalmente, manejan variables de 16 y 32 bits. Las cuales, necesitan una unidad aritmético lógica (ALU por sus siglas en inglés *Artimetic-Logic Unit*) más robusta que un microcontrolador normal, ya que son diseñados con características que permiten procesar tareas complejas, repetitivas y numéricamente intensas [41].

3.1.5. Ordenador de placa reducida

Es un ordenador donde toda su circuitería se realiza en una sola placa. Con ello, sus dimensiones pueden ser miniaturizadas. No obstante, carece de partes elementales instaladas por defecto como una pantalla, cámara, teclado, entre otros. Sin embargo, cuenta con los periféricos suficientes para ofrecer estas prestaciones. Actualmente, los recursos de velocidad y almacenamiento computacional han evolucionado significativamente. Por esta razón, su principal diferencia con un sistema embebido, se basa en su aplicabilidad, ya que son sistemas multipropósito que se puede instalar una gran cantidad de *software* para generar distintas aplicaciones de forma simultánea [50].

3.1.6. Pruebas de software y hardware embebido

Son estrategias planteadas para el aseguramiento de la calidad del proceso a cumplir mientras se realizan pruebas de detección y seguimiento de errores. Además, se pueden encontrar las debilidades de *software* y *hardware*. Para su validación, se plantean dos técnicas importantes. Por un lado, el `análisis de mutaciones` que introduce fallos intensionados para observar el comportamiento del sistema. Por otro lado, el `análisis de estados de transición`, que se enfoca en verificar las relaciones entre los eventos, acciones y estados del sistema [49].

3.1.7. Modelo de desarrollo de sistemas smbebidos

El desarrollo de sistemas embebidos se basa en dos componentes importantes. El primero es el *hardware*, ya que es el encargado de la recopilación de datos mediante la integración de todos los componentes electrónicos. Para ello, definir correctamente su aplicación permitirá un adecuado desarrollo de un prototipo para su etapa de validación y, posteriormente, pueda ser empleado en condiciones reales. El segundo, es el *software*, encargado de crear la funcionalidad al sistema embebido que le permita comunicarse con todos los componentes de *hardware* [43]. En algunos casos, este componente tiene la tarea de buscar la forma de interactuar con el ser humano. Como resultado, ambos componentes deben trabajar en conjunto de manera óptima. Sin embargo, los diferentes estándares se han desarrollado en forma diferenciada [39]. No obstante, con el avance del aprendizaje automático, es necesario buscar la manera de integrar ambos conceptos en uno solo. Por esta razón, el presente modelo se basa la metodología de desarrollo `V` de *software* debido a sus etapas separadas y secuenciales que pueden ser validadas entre ellas y el estándar `IEEE 29148`, que establece la forma de definir los requisitos y funcionalidades de *hardware*. En la Fig. 3.2 se muestra el modelo propuesto.

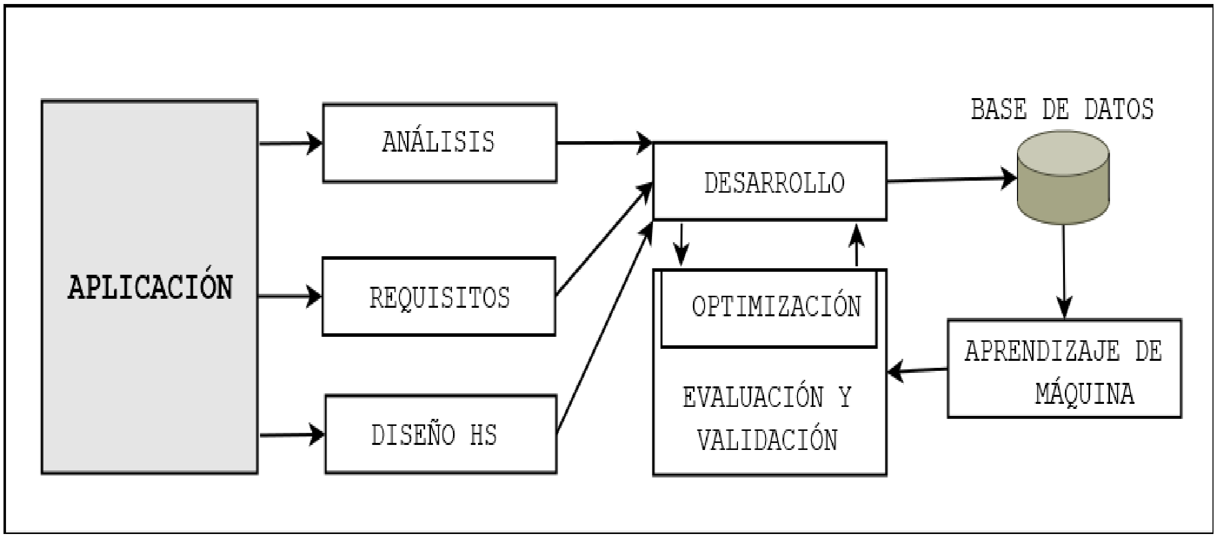


Figura 3.2: Modelo propuesto de desarrollo de sistemas embebidos

3.2. Sensores

La mayoría de las variables existentes en el mundo real deben medirse con un dispositivo que convierta los fenómenos en una forma que el humano pueda percibir con sus sentidos. Estos dispositivos encargados de este proceso se denominan sensores. Ya que es un elemento electrónico capaz de convertir un fenómeno físico a una señal eléctrica, para ser procesada y analizada. En este sentido, el sensor cuenta principalmente con dos componentes. Por un lado, se necesita un elemento sensitivo que permite realizar la interacción con el medio y que tiene la posibilidad de variar su valor para la correspondiente adquisición de los datos. Por otro lado, se debe contar con un transductor. Éste es el encargado de convertir dicha medición a una señal eléctrica. En consecuencia, los sensores pueden clasificarse según la naturaleza del fenómeno a ser empleado, la forma de adquisición del sistema, el tipo de construcción del sensor, entre otros [49, 51].

3.2.1. Métricas de evaluación de sensores

Las métricas relcionadas con el desempeño con sensores, permiten conocer la capacidad de recrear el fenómeno estudiado a partir de los componentes almacenados de la señal [49]. Los más relevantes son:

3.2.1.1. Exactitud

Representa la exactitud del valor de salida del sensor hacia el sistema electrónico en comparación con el valor real.

3.2.1.2. Precisión

La capacidad del sensor de proporcionar la misma lectura al realizar repetidas veces el mismo experimento. Es un valor estadístico que puede evaluarse mediante la desviación estándar.

3.2.1.3. Repitibilidad

Es la capacidad de reproducir las mismas lecturas desde en un etorno controlado.

3.2.1.4. Reproducibilidad

Es la capacidad del sensor de repetir las mismas respuestas después de haberse alterado las condiciones de medición.

3.2.1.5. Estabilidad

La facultad de producir el mismo valor de salida en pruebas controladas de su funcionamiento en periodos de tiempo similares.

3.2.1.6. Ruido

Es el cambio en la salida del sensor cuando no está expuesto a ningún estímulo.

3.2.1.7. Error

Se denominan a las fluctuaciones de la señal existentes, al no modificar el valor de entrada medido. Este es un factor muy importante sobre la calidad de los datos extraídos por medio del sensor. La relación señal/ruido es usada en aplicaciones con sensores con la forma $\frac{S}{N}$ donde S es el promedio de la señal y N es la desviación estándar del ruido [52].

3.2.1.8. Resolución

Es el valor mínimo que se puede detectar en la variación del estímulo. Está limitada fuertemente por ruido de la señal y la capacidad de procesamiento que cuenta el sistema embebido.

3.3. Adquisición de datos

Generalmente, las señales eléctricas adquiridas por un sensor pueden ser observadas con aparatos tradicionales de medida como voltímetros o amperímetros. No obstante, la creciente necesidad de registrar y preservar estos fenómenos para analizar datos en posterior, obliga a desarrollar sistemas que permitan realizar este proceso de una forma flexible con respecto a los equipos antes mencionados que no lo pueden realizar. Anteriormente, se empleaban sistemas que eran considerados como tarjetas complementarias a un computador. Sin embargo, con el desarrollo de los sistemas embebidos, se volvieron independientes y capaces de adquirir datos de gran volumen por su propia cuenta.

Estos sistemas embebidos buscan contar con datos confiables, precisos, repetibles y sin errores. Por esta razón, es necesario seleccionar adecuademente los sensores para cada aplicación, capturar la señal con la debida frecuencia, rango y magnitud y realizar un impedancia correcta para la interconexión entre sistemas. Ya que la naturaleza de las señales son de carácter no lineal y eso hace más difícil su tratamiento. Además, tomando en consideración que estos sistemas pueden ser susceptibles a golpes, temperaturas extremas, vibraciones, desgaste del materiales, entre otros [1, 49].

Cuadro 3.1: Tipos de acondicionamiento de datos de sensores [2]

Tipo	Descripción
Amplificación	Incrementan el nivel de voltaje para que sea el adecuado hacia el conversor analógico-digital (CAD).
Atenuación	Se necesita reducir el nivel de voltaje para no dañar el CAD ya que excede su rango de trabajo.
Filtrado	Las señales son propensas a interferencias, para ello se utilizan filtros para eliminarlas.
Excitación	Algunos sensores necesitan de una fuente de alimentación externa para funcionar por su alto consumo de corriente.
Linealización	Cuando la señal que produce el sensor no está linealmente relacionada con la medida física que mide.

No obstante, ciertos sensores cuentan con diferentes etapas de acoplamiento y, en algunos casos, con microcontroladores para mejorar el proceso de adquisición de datos. En este sentido, los datos ya no provienen de una señal análoga a ser procesada. En vez de ello, cuentan con una interpretación en un lenguaje digital, que puede ser enviado hacia el sistema embebido como trenes de pulsos o estados lógicos. Por este motivo, la sección 3.3.1 y 3.3.2 muestra el acondicionamiento analógico y la sección 3.3.3 el acondicionamiento digital. Este proceso se puede resumir en la Tabla 3.1.

3.3.1. Acondicionamiento analógico

La información obtenida por el sensor es variante en el tiempo y suceptible a ligeras variaciones en su magnitud que debe detectar el sistema embebido. Por tal motivo, debe existir un proceso de cuantificación y codificación unívoca binaria. Donde cada valor binario, corresponde a un solo valor de tensión o corriente. Como resultado, una señal analógica es convertida a una señal digital para ser tratada. No obstante, este proceso puede producir pérdidas de información. Por esto, la resolución (escala capaz de representar una señal análoga) juega un papel importante en este proceso.

En un sistema embebido, se cuenta con varios canales CAD con una resolución entre 8 y 32 bits, los cuáles pueden ser [2, 9]:

- Aproximación sucesiva: Es el *hardware* comúnmente utilizado en microcontroladores por su facilidad de implementación y flexibilidad en los muestreos de la señal. El cual, usa comparadores por medio de amplificadores operacionales para codificar la señal. En la Fig. 3.3 se muestra su estructura interna

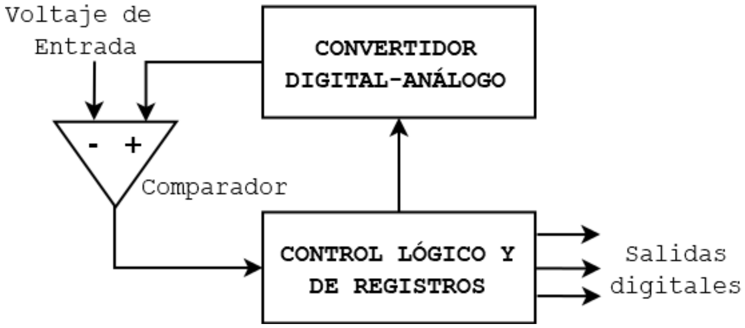


Figura 3.3: Diseño interno del conversor análogo-digital de aproximación sucesiva [1]

- Voltaje a frecuencia: Esta forma de CAD se encuentra orientado a variar su frecuencia al recibir cambios de voltaje de entrada al sensor. Generalmente, se utiliza en sensores digitales que envían los datos por medio de protocolos de comunicación cableada. Además, necesita de un circuito de reloj que permita sincronizar los datos desde la entrada de la señal hacia la generación de pulsos de diferentes frecuencias. En la Fig. 3.4, muestra la estructura interna.

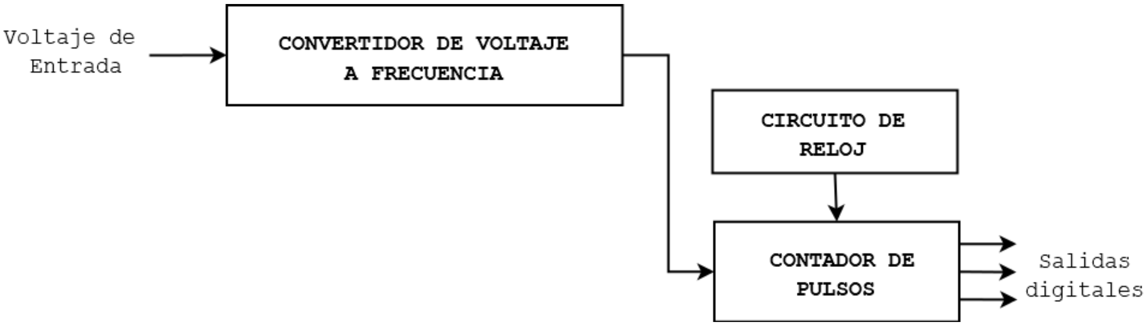


Figura 3.4: Diseño interno del conversor análogo-digital de voltaje a frecuencia [1]

- Sigma-Delta: Es la estructura de mayor complejidad desarrollada, debido a la integración de filtros digitales que mejoran el proceso de adquisición de datos. A pesar de que son muy robustos, su complejidad de uso y su escasa flexibilidad no permiten emplearlos fácilmente. Por esta razón, están diseñados para aplicaciones específicas [1]. En la Fig. 3.5 se muestra su estructura interna.

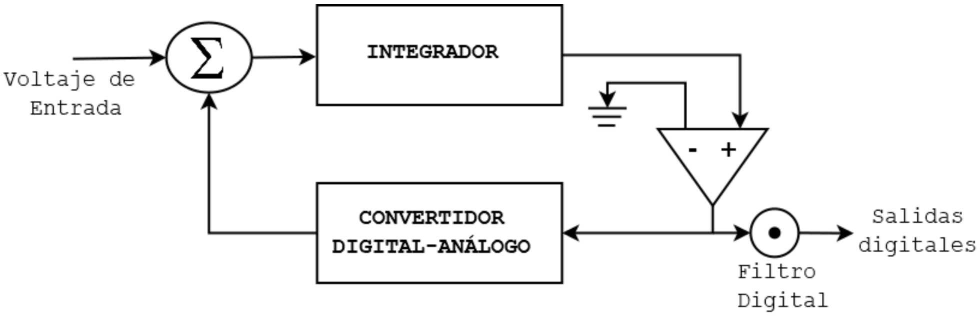


Figura 3.5: Diseño interno del conversor análogo-digital sigma-delta [1]

- Integrador: Es el proceso más sencillo de realizar una conversión análoga digital al usar el tiempo de carga y descarga de un capacitor. Sin embargo, se encuentran descontinuados en la actualidad por ser muy susceptibles a fallos [1]. En la Fig. 3.6 se muestra el funcionamiento interno de un capacitor al tener un señal análoga y ha sido integrada.

3.3.2. Filtros

Todos los sistemas de adquisición de datos son susceptibles a tener errores de lecturas. Con ello, los datos pueden perder su validez y usabilidad. Por estas razones, la implementación de filtros permite, por un lado, restaurar la señal mediante los algoritmos de suavizado de señal. Para ello, se necesita tener componentes pasadas y futuras. Por otro lado, se pueden separar componentes no deseadas que son relacionadas al ruido. Esta modificación de la señal permite ser entendida de mejor manera. Para hacer esto, se emplean sistemas lineales invariantes en el tiempo (LTI por sus siglas en inglés *Linear Time-Invariant*). Éstos permiten tener una señal de entrada discreta $X[n]$ que al aplicarse a una función de transferencia $H[n]$, se obtiene una señal de salida $Y[n]$ mejorada. Donde $H[n]$ puede ser en respuesta a la frecuencia o al escalón (dominio del tiempo) [2]. Cada uno

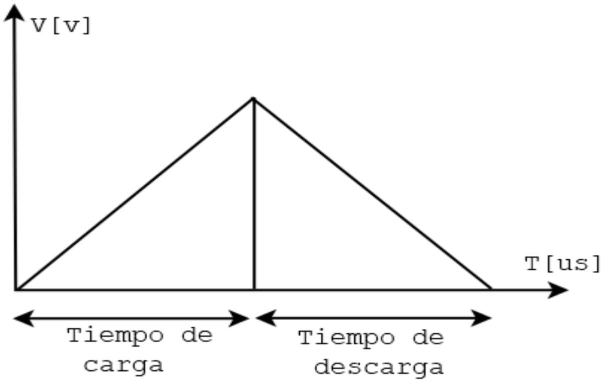


Figura 3.6: Diseño interno del conversor análogo-digital integrador [1]

de ellos cuentan con parámetros independientes que se detallan a continuación:

3.3.2.1. Respuesta en frecuencia

Este análisis se origina al observar sus componentes de energía en un amplio rango de frecuencias que contiene la señal. Para hacer esto, es importante conocer la frecuencia de muestreo que ha sido adquirida la señal y el origen de la misma para tener un conocimiento teórico de sus frecuencias principales y las que generalmente pueden ser consideradas como componentes de ruido [2]. Con esto, se puede diseñar un filtro pasa bajos, altos, banda y rechaza banda. Las principales características de los filtros en frecuencia son:

- Ganancia: Capacidad de amplificar la energía de la señal de entrada para ser observada de mejor manera.
- Frecuencia de corte: Es la frecuencia donde inicia el filtro a eliminar las componentes no deseadas.
- Región de transición: Es la zona donde se presenta la pendiente de caída de ganancia, que va desde la frecuencia de corte del filtro, hasta la banda a ser eliminada. En esta zona, la pendiente del filtro puede variar dependiendo del tipo de aproximación y el orden del filtro (capacidad de atenuar una señal) que se ocupe en el momento de diseñarlo.
- Banda eliminada: Es el rango de frecuencias que se encuentra después de la región de transición, donde la ganancia es mínima, por lo tanto, su energía de salida tiende a cero.

3.3.2.2. Respuesta en el tiempo

La respuesta al escalón, es el análisis de la señal desde el dominio del tiempo. Este proceso es realizado para los cambios entre regiones no similares, por ejemplo, cuando un evento empieza o termina [2]. Para el diseño del filtro, se deben considerar las siguientes características:

- Velocidad de trancisión (*risetime*): Es el tiempo que se demora el filtro entre la banda eliminada y la banda de paso
- Cambio en la amplitud del escalón (*overshoot*): Los picos de amplitud entre la banda eliminada y la banda de paso
- Simetría entre las mitades superior e inferior del pulso: Representa la linealidad de fase (la capacidad de atenuar las señales desde la banda de transición).

Estas dos formas de representar una señal, derivan en los tipos de filtros que se pueden emplear. En consecuencia, un filtro de respuesta finita al pulso (FIR por sus siglas en inglés Finite Impulse Response), se trata de un análisis a una señal impulso (criterio común de diseño de filtros) como señal de entrada, tendrá finitos términos de salida. Sin embargo, un filtro de respuesta infinita al impulso (IIR por sus siglas Infinity Impulse Response), si la señal de entrada es un impulso, tendrá infinitos términos no nulos, es decir, cuenta con retroalimentación y por tal motivo, no puede volver al reposo [53].

3.3.2.3. Finita al impulso (FIR)

Se basan en el criterio de la convolución. Son muy estables ya que son realizados con ceros en el plano complejo. Generalmente, son implementados en *software* donde emplean funciones *kernel* en el dominio del tiempo (considerados como ventanas), ya que son funciones matemáticas que tiene un valor de cero fuera del intervalo escogido. Por esta razón, pueden actuar como filtros para eliminar componentes que salgan de su rango. Estas funciones, usualmente tienen valores máximos en la posición central de la señal y disminuye al alejarse de ella [53]. En este sentido, las funciones relevantes son: *Barlett, Hamming, Kaiser, Nutall y Gaussiano* [53].

A pesar que en su forma puedan ser muy similares, cada una de las ventanas al modificar sus parámetros de diseño del filtro, pueden adaptarse a la forma de la señal de entrada. En la Fig. 3.7 se muestra las ventanas en el dominio del tiempo con sus valores estándar de filtrado a 50 Hz pasa-bajos.

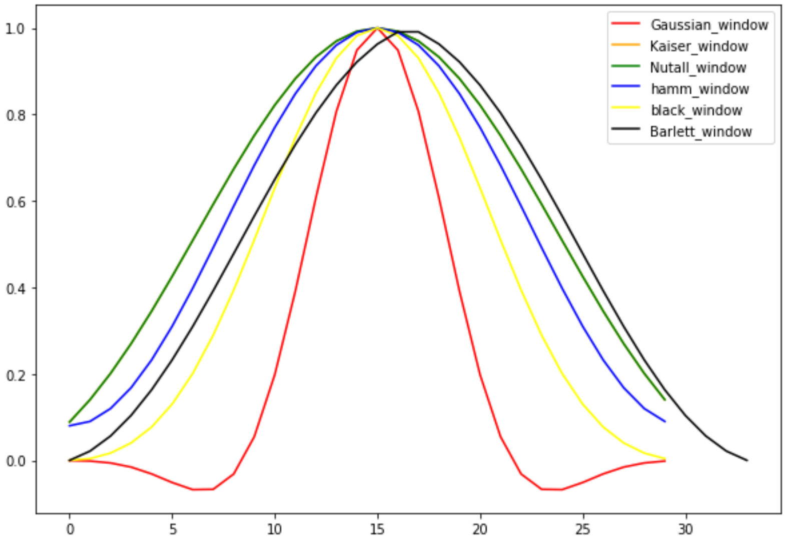


Figura 3.7: Funciones kernel para filtros FIR

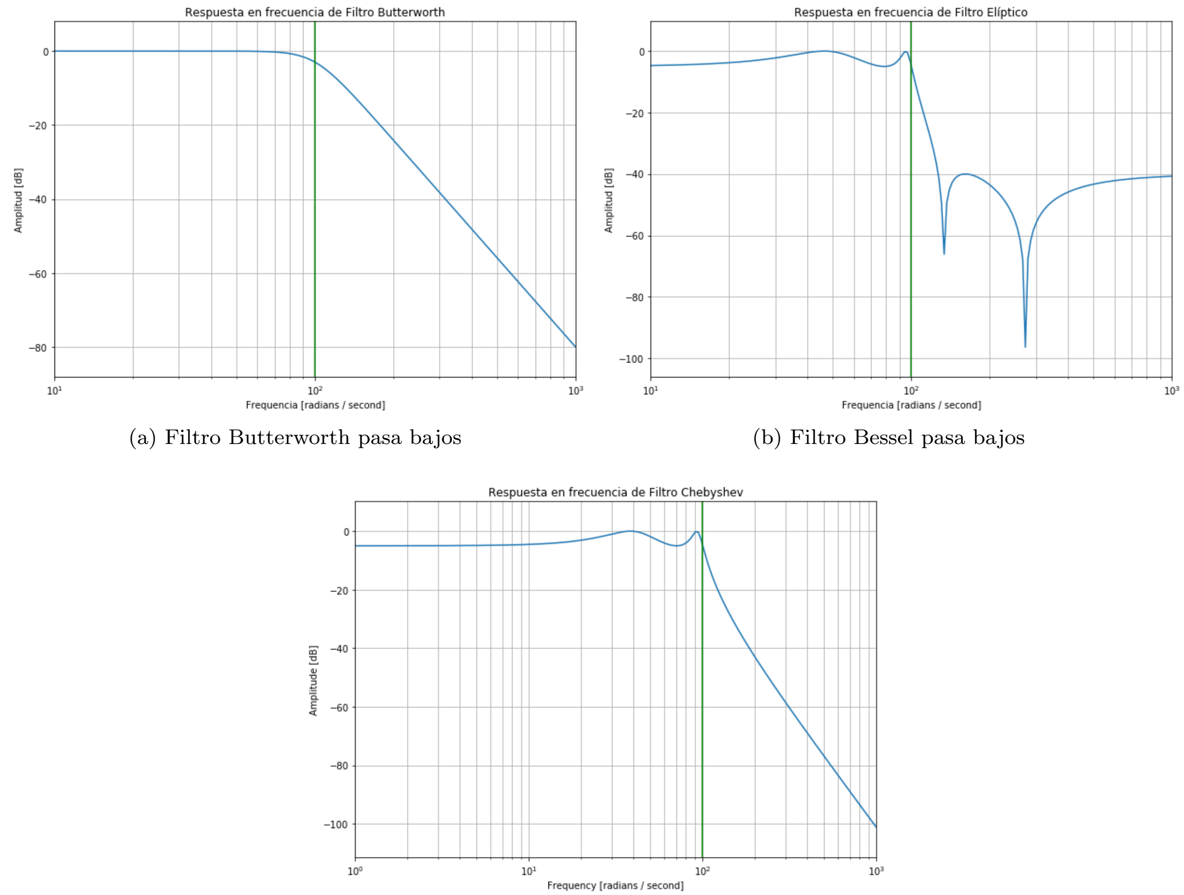
3.3.2.4. Infinita al impulso (IIR)

Estos filtros a pesar que tienen como requisito la retroalimentación, no requieren demasida memoria y su codificación es sencilla. Sin embargo, cuenta con polos y ceros en el plano complejo. Por esta razón, pueden ser inestables. Sus implementaciones están en relación a aproximaciones, que son [53]:

- Butterworth: Es una aproximación para filtros electrónicos diseñada con el objetivo de contar con una respuesta plana, la banda de transición sea la menor posible y una banda eliminada con mayor pérdida.

- Bessel : Esta aproximación tiene una región de transición adecuada. Sin embargo, no cuenta con una banda de respuesta plana.
- Chebyshev: Cuenta con la mayor pérdida en las bandas eliminadas pero genera rizos (variaciones de energía/voltaje) en su banda pasante. Sin embargo, al aumentar el filtro, estos errores tienen a disminuir.

De acuerdo con sus características, se aprecia de forma gráfica su respuesta en frencuencia, como se muestra en la Fig. 3.8.



(a) Filtro Butterworth pasa bajos

(b) Filtro Bessel pasa bajos

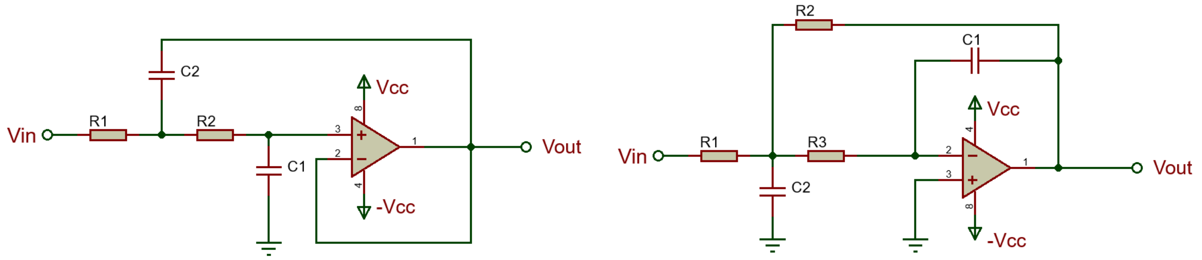
(c) Filtro Chebyshev pasa bajos

Figura 3.8: Respuesta en Frecuencia de filtros IIR

Este tipo de aproximaciones pueden ser implementadas en *hardware* por medio de estructuras definidas. Esto quiere decir que su diseño es estándar en la ubicación de resistencias y capacitores y en relación a la apróximacion a emplear, solo cambian sus valores. En este sentido, éstas pueden ser:

- Sallen-Key: Es un tipo de filtro electrónico activo particularmente valioso por su simplicidad. El circuito produce un filtro pasa bajo o pasa alto de dos polos usando dos resistencias, dos condensadores y un amplificador. Para obtener un filtro de orden mayor, se agregan en forma secuencial varias etapas [1].
- Multiple Feedback (MFB): A diferencia del anterior, MFB posee el sistema retroalimentación para mejorar la etapa de ganancia [1].

Sus correspondientes circuitos se muestran en la Fig. 3.9



(a) Estructura *Sallen-Key*

(b) Estructura *Multiple Feedback*

Figura 3.9: Circuitos de cada estructura de filtros IIR, V_{in}: Voltaje de entrada, V_{out}: Voltaje de salida, V_{cc}: Alimentación de los amplificadores operacionales [1]

3.3.2.5. Suavizado de la señal

El suavizado de la señal puede dar mejores resultados con respecto al ruido. Ya que se basa en realizar procesos matemáticos entre las mismas componentes para diferenciar las menos comunes y ser eliminadas. Sin embargo, requiere muestras pasadas y futuras de la señal que en algunos casos no siempre es accesible. No obstante, el suavizado mejora la calidad de los datos al reemplazar la señal irregular ruidosa con la nueva señal suavizada que probablemente describe mejor los fenómenos medidos [54]. Los algortimos más usados con respecto a una señal $x[i]$ de i muestras con una ventana de tamaño n, para obtener una señal de salida $y[i]$ son:

- Filtro Promedio:

$$y[i] = \frac{\sum_{i-n}^{n} x_j + i}{2n+1} \tag{3.3.1}$$

- Filtro Medio:

$$y[i] = median[x_{i-n}, ..., x_i, ..., xi + n] \tag{3.3.2}$$

- Filtro Savitzky-Golay:

$$y[i] = (2d+1)^{-1} \sum_{i=n-d}^{n+d} x[i], \tag{3.3.3}$$

- Filtro de Kalman:

$$\frac{d}{dt} y[i] = A[n]x[i] + B[n]u[i] \tag{3.3.4}$$

- Filtro Gaussiano:

$$y[i] = \sqrt{\frac{n}{\pi}} \cdot e^{-a \cdot x[i]2} \tag{3.3.5}$$

3.3.3. Acondicionamiento digital

En algunas aplicaciones digitales, la frecuencia de pulsos se compara en una base de tiempo fija. Los circuitos monitorean el tren del pulsos, condicionando para un nivel adecuado entre el estado alto-bajo. Con ello, se evitan los rebotes del sistema y se mejora la capacidad de adquirir datos a altas velocidades. Esto se realiza mediante un comparador que genera un ancho de pulso constante cada vez que la señal de entrada pasa por cero. Luego, el pulso pasa a través de un circuito integrador que genera un nivel de señal que cambia lentamente en su salida, proporcional a la frecuencia de entrada [1]. Como se había comentado, este criterio es considerado cuando el propio sensor ya ha realizado el proceso de convertir la señal a un lenguaje digital.

3.4. Algoritmos de aprendizaje supervisado

La tendencia tecnológica actual frecuentemente implica la toma de decisiones por medio de la comprensión de grandes cantidades de datos, que requerirían de mucho tiempo, para almacenar y analizar de forma tradicional. Las técnicas de aprendizaje automático proporcionan herramientas útiles para el diseño de sistemas que extraen conocimiento de los datos. Hablamos de aprendizaje automático o *machine learning* [55], cuando los sistemas aprenden y se adaptan a los cambios de forma autónoma, proporciona soluciones para todos los problemas posibles. Pudiendo de forma automatizada identificar patrones o tendencias. El aprendizaje automático utiliza diversas categorizaciones de sus algoritmos con diferentes criterios que se han probado su eficiencia para la resolución de diferentes problemas.

Teniendo en cuenta la naturaleza de los datos de entrenamiento, las técnicas de aprendizaje automático que se utilizan en minería de datos permiten diferenciar entre aprendizaje supervisado y no supervisado [56]. En el aprendizaje supervisado, los algoritmos trabajan con datos etiquetados o valores numéricos. Lo que permite conocer de antemano, la etiqueta o clase o el valor de predicción que le corresponde a cada instancia del conjunto de entrenamiento. El objetivo, en este caso, es aprender de los datos utilizados en la fase de entrenamiento, los patrones que presentan los datos para las etiquetas que se consideran y buscar dichos patrones en cada instancia de los datos de prueba para asignarle la etiqueta o el valor más adecuado. Estos algoritmos se utilizan comúnmente en problemas de regresión y clasificación. Por otra parte, el aprendizaje no supervisado consiste en el análisis de datos no etiquetados con el objetivo de realizar agrupaciones de las diferentes instancias basadas en sus similitudes. A partir de esta categorización de los algoritmos existen otras muchas técnicas de aprendizaje como el semi-supervisado y el aprendizaje por refuerzo.

El concepto de aprendizaje automático comprende una inmensa cantidad de técnicas y algoritmos que no podemos tratar en su totalidad en esta tesis doctoral. Por ello, nos enfocaremos de manera general sobre las técnicas más apropiadas para la resolución de los problemas que se van a abordar y de manera específica sobre los algoritmos de aprendizaje supervisado que se van a aplicar en este trabajo. El objetivo del estudio es seleccionar los más adecuados para la búsqueda de conocimiento útil a través de los sistemas embebidos, con el fin de que puedan tomar sus propias decisiones sin la interacción con el ser humano [56].

Como hemos dicho, el aprendizaje automático puede proporcionarnos los métodos para obtener conocimiento de un conjunto de datos, en casos donde las personas no pueden hacerlo debido a la cantidad y complejidad de la información [43, 57]. No obstante, a pesar de que estas técnicas son muy empleadas, el tiempo de ejecución, las restricciones de memoria y el uso de librerías complejas, pueden ocasionar que no todos los sistemas puedan contar con la capacidad para implementarlos. Con más razón, si se emplean sistemas embebidos que no cuentan con recursos computacionales suficientes, comparados con un ordenador estándar.

3.4.1. Características de los datos

Las características o atributos (*features*) de los datos son términos relevantes que permite encontrar patrones y poder entrenar el clasificador. Por esta razón, se busca una metodología de selección que pueda ser empleada dentro de un sistema embebido para la obtención de datos adecuados para entrenar un modelo de clasificación supervisada. Para ello se establecen tres etapas: selección de prototipos, reducción de la dimensionalidad y selección de características [58].

3.4.1.1. Selección de Prototipos

Las técnicas de selección de prototipos (SP) se basan en el concepto de que no todos los datos proporcionan información relevante para el clasificador. Por esta razón, su primer objetivo es reducir la base de datos de entrenamiento así como aumentar o mantener un rendimiento de clasificación. Existen tres criterios al aplicar la selección de prototipos: (i) `Condensación:` incluye técnicas con el objetivo de retener los puntos más cercanos a los límites de decisión, conocidos como puntos de borde. Esto se debe a la observación de que los puntos internos no afectan los límites de decisión como los puntos fronterizos y, por lo tanto, pueden eliminarse con un efecto relativamente pequeño en la clasificación. (ii) `Edición:` busca eliminar los puntos de borde. Ya que son considerados como datos ruidosos o que no están de acuerdo con sus vecinos, dejando los límites de decisión más suaves. Finalmente, (iii) `Híbrido:` intentan encontrar el subconjunto más pequeño denominado S que mantiene o incluso aumenta la precisión de la generalización en la prueba de datos. Para lograr esto, permite la eliminación de los puntos internos y los puntos de borde de acuerdo con los criterios seguidos por las dos estrategias anteriores [57]. En la Tabla 3.2, se muestran los algoritmos más representativos de cada criterio.

Cuadro 3.2: Algoritmos de SP

Método	Nombre	Abreviatura
CONDENSACIÓN	Condensed Nearest Neighbor Reduced Nearest Neighbor Selective Nearest Neighbor	CNN RNN SNN
EDICIÓN	Edited Nearest Neighbor All-k Edited Nearest Neighbors Iterative Partitioning Filter	ENN AENN IPF
HÍBRIDO	Decremental Reduction Optimization Procedures 2 Decremental Reduction Optimization Procedures 3 Iterative Noise Filter based on the Fusion of Classifiers	DROP2 DROP3 INFFC

3.4.1.2. Reducción de la dimensionalidad

Algunos criterios de aprendizaje supervisado, mencionan que mientras mayor sea el tamaño de los datos adquiridos, mejor será la información que obtendremos de ellos. No obstante, este fundamento puede ser suceptible al presentar el conocimiento intrínseco de los datos en el desarrollo de interfaces de usuario. Ya que esto se basa en la percepción humana, que hace uso de los sentidos de las personas, especialmente la visión. Esta, es considerada como el medio entre la información obtenida observada desde un computador y el cerebro humano. En este sentido, nuestra visión tiene la capacidad de percibir datos hasta en tres dimensiones, limitando de esta forma a la presentación de resultados por parte de un algoritmo de aprendizaje automático. Es por ello, que los algoritmos de reducción de dimensionalidad (RD), pretenden reducir la complejidad de los datos y en consideración con el rendimiento de respuesta de un sistema embebido, logran reducir el número de variables a uno menor. Esto se realiza con el objetivo de tener una mayor compresión visual del proceso realizado por el algoritmo y reducir el tamaño de la información almacenada a ser procesada [59,60].

Los métodos usados para la reducción de dimensionalidad son: espectrales, disimilitudes, basados en divergencias y heurísticos [61]. Por su parte, [62] afirma que los métodos espectrales como

LLE, LE, el método lineal como PCA y métodos estocásticos basados en divergencias como SNE y t-SNE son los principales y la base de todos los métodos existentes de RD. Por esta razón, se describen solo estos algoritmos a continuación:

- Principal Component Analysis (PCA): es un método que busca conseguir un nuevo conjunto de variables o componentes principales que están incorrelacionadas entre sí, a través de transformaciones ortogonales realizadas al conjunto de variables originales. Todo esto se realiza con el objetivo de reducir la dimensionalidad de estos. Una característica de esta técnica es lograr trabajar fácilmente con gran cantidad de datos, consiguiendo así evitar una gran carga computacional [62].

- Laplacian Eigenmaps (LE): es un algoritmo que utiliza técnicas espectrales, es decir, se basa en la suposición de que los datos se encuentran en un forma de baja dimensión en un espacio de alta dimensión. Donde cada dato se utiliza como un nodo y la conectividad entre ellos se rige por la proximidad de los puntos vecinos [63].

- Locally Linear Embedding (LLE): busca hallar la variedad de baja dimensión dentro del conjunto de datos que son de alta dimensión. Además, pretende conservar la propiedad de que si dos puntos estaban próximos en el espacio origen, lo seguirán estando en el nuevo espacio de baja dimensión. Para ello, se realiza una búsqueda de vecinos más cercanos para construir la matriz de pesos y descomponer parcialmente los valores propios [63].

- Stochastic Neighbor Embedding (SNE): utiliza una distribución gaussiana para cada valor de entrada de los datos de alta dimensión con el fin de usar su densidad. De esta manera, establece una distribución de probabilidad de todos los vecinos. Luego, aproxima esta distribución de probabilidad tanto como sea posible repitiendo la estructura de parentesco en un espacio de menor dimensión [63].

- T-distributed Stochastic Neighbor Embedding (t-SNE): es un algoritmo que hace uso de una distribución que mide la similitud entre pares de objetos de entradas y una medida de distribución entre parejas similares de los correspondientes puntos de análisis, representando así un conjunto con una menor dimensión. Este método comienza por encontrar patrones en los datos mediante la identificación de grupos que han sido observados según la similitud de los puntos de datos con varias características. Realiza la reducción de dimensionalidad debido a que asigna los datos multidimensionales a un espacio de menor dimensión [62].

Todos ellos, cuentan con algoritmos relevantes, que se presentan en la tabla 3.3.

Cuadro 3.3: Algoritmos de RD

Método	Nombre	Abreviatura
ESPECTRAL	Locally Linear Embedding	LLE
	Laplacian Eigenmaps	LE
BASADOS EN DISIMILITUDES	Principal Component Analysis	PCA
	Multidimensional Scaling	CMDS
DIVERGENCIAS	Stochastic Neighbor Embedding	SNE
	T-Distributed SNE 3	t-SNE
HEURÍSTICOS	Self-Organizing Maps	SOM
	Análisis de Componentes Curvilíneos	CCA

3.4.1.3. Selección de características

El tiempo de respuesta en un sistema embebido puede ser crucial para las diferentes aplicaciones que pueden emplearse [64]. Normalmente, para una tarea de clasificación, una gran cantidad de atributos pueden ser candidatos para fines de caracterización. Sin embargo, muchos de estos atributos pueden ser irrelevantes o redundantes y, en consecuencia, el algoritmo de clasificación puede sufrir un sobreajuste. Por esta razón, algunas características importantes del aprendizaje supervisado pueden verse afectadas. Por lo tanto, la clasificación podría no alcanzar el rendimiento esperado [65]. De hecho, en muchas aplicaciones que involucran grandes conjuntos de datos, los clasificadores no funcionan correctamente hasta que las características no deseadas se eliminan obligatoriamente [64].

En este sentido, la tarea de selección de características permite reducir el número de atributos que representan un conjunto de datos y que se ajustan a un modelo o tarea de aprendizaje automático. Existe una amplia gama de criterios y algoritmos que se pueden utilizar para esta tarea. Sin embargo, dependiendo de la naturaleza y el objetivo del análisis de datos, algunos criterios de selección de características pueden resultar más adecuados. En este trabajo, la tarea de selección de características se lleva a cabo mediante un enfoque que utiliza dos criterios. Por un lado, la correlación de variables que emplea un análisis basado en la probabilidad bajo una distribución chi-cuadrado. Por otro lado, el método de filtrado denominado `RelieffF` se utiliza para la selección de características en sí, que es una técnica dirigida por variables objetivo [66]. Cabe recalcar, que cada criterio puede representar de forma porcentual el aporte de cada variable al clasificador.

3.4.2. Clasificación supervisada

Desde el punto de vista de clasificación supervisada, el objetivo principal consiste en asignar a un objeto o fenómeno físico a una categoría o clase previamente reconocida. Para ello, es necesario contar con un conjunto de datos descritos por un vector de características y a la clase que pertencen cada uno de ellos (datos de entrenamiento). De esta forma, se construye el modelo o regla que se utilizará para clasificar nuevas instancias (datos de prueba). Además, para estimar el rendimiento del clasificador (medir la capacidad de una predicción correcta), se debe medir su tasa de error, la rapidez con la que el clasificador construye el modelo, la simplicidad del modelo y la interpretabilidad [55].

No obstante, dado que existen diferentes algoritmos, es necesario determinar el apropiado que pueda presentar al conjunto de datos adquirido previamente. Basado en un entrenamiento depurado, la tarea principal del sistema es la identificación adecuada de un fenómeno estudiado. Debido a esto, los criterios de clasificación han sido tomados por: (i) distancias, (ii) probabilidades, (iii) siguiendo un modelo, (iv) heurística, (v) hipótesis y (vi) aprendizaje profundo para determinar el apropiado para sistemas embebidos [56].

3.4.2.1. k-Vecinos cercanos

Se considera el algoritmo basado en distancias denominado k-Vecinos Cercanos (k-NN por sus siglas en inglés *k-Nearest Neighbors*), que asocia una nueva instancia con la base de entrenamiento para asignar al grupo más cercano de acuerdo con su distancia más corta. Donde k es el número de vecinos a tomar en cuenta para la asignación de etiqueta. Según la literatura, los mejores resultados se obtienen con k = 1 y con k = 3 [56].

3.4.2.2. Clasificador Bayesiano

El enfoque bayesiano para clasificar la nueva instancia consiste en asignar el valor objetivo más probable $P(Ci/E)$ dados los valores de los atributos que describen dicha instancia, tal y como se puede observar en la Ecuación 3.4.1 [56].

$$P(C_i|E) = P(C_i)P(E|C_i)/P(E). \tag{3.4.1}$$

3.4.2.3. Árboles de decisión

Como criterio heurístico, se utilizan los árboles de decisión. Dado que un clasificador se puede definir como una función d($\boldsymbol{x}$), que divide el espacio de entrada original $\underset{\sim}{X}$ en M diferentes subconjuntos $\{\boldsymbol{A}_1, \ldots, \boldsymbol{A}_M\}$, de modo que $\boldsymbol{X} = \cup\,\mathrm{lím}_{m=1}^{M} \boldsymbol{A}_m$, y puede predecir las clases correspondientes C_m. En otras palabras, dicho algoritmo busca dividir el espacio de clasificación en zonas, de modo que los patrones que pertenecen a cada zona se asignen a una de las clases conocidas [56].

3.4.2.4. Máquinas de soporte de decisión

Según los criterios basados en modelos, se utiliza el algoritmo de máquinas de soporte decisión (SVM de sus siglas en inglés *Support Vector Machines*). Ya que cuenta con funciones matemáticas que permiten evidenciar de mejor forma los bordes decisión del conjunto de datos. Con esto, puede funcionar con cualquier tipo de variables ya que se basa en la búsqueda de un espacio de características efectivo. Para hacer esto, se usan los métodos *kernel*, que se puede considerar como un hiperplano óptimo de separación que se denota como el producto interno. Estas funciones pueden ser: lineal, polinómica, gaussiana o sigmoide [67].

3.4.2.5. Regresión Logística

Según el criterio de hipótesis, la regresión logística es un tipo de análisis utilizado para predecir el resultado de una variable categórica a partir de variables independientes o predictivas que siguen una relación lineal [68]. En términos generales, dicha relación lineal se puede escribir:

$$g(E(y)) = \alpha + \beta x_1 + \lambda x_2, \tag{3.4.2}$$

Donde $g(\cdot)$ es la función de enlace, $E(y)$es la expectativa de la variable objetivo y $\alpha + \beta x_1 + \lambda x_2$ es el modelo predictor lineal, siendo el α, β, y λ los parámetros a predecir [68]. Sin embargo, este criterio solo puede ser empleado en clasificación de dos clases. Conviertiéndose en un método muy restrictivo.

3.4.2.6. Aprendizaje profundo

Finalmente, se considera a una red neuronal como un modelo de propagación directa de información que permite conocer las características del conjunto de datos sin la necesidad de conocer adecuadamente sus atributos. Hacen referencia al funcionamiento del cerebro humano que aprende a base de estímulos, donde cada neurona reacciona sobre el evento presentado y se asocia con otras para compartir información. Para ello, emplean nodos de activación que se ejecuta como un interruptor. Es decir, activan a un grupo de neuronas para generar un valor de salida a partir de un valor de entrada en un rango establecido. Para ello, como recomendación en un diseño de red neuronal, las funciones de activación en las capas ocultas deben ser lineales $y = x$ y la capa final con la función de activación sigmoidea $(\mu, x) = \frac{1}{1-e^{-x}}$ [69]. La propagación de funciones parciales y finales para cada out_k se puede expresar como:

$$\begin{cases} out_h^{(0)} &= in_h, \\ out_i^{(1)} &= f(\sum_h out_h^{(0)} w_{hi}^{(1)}), \\ out_j^{(2)} &= f(\sum_i out_i^{(1)} w_{ij}^{(2)}), \text{ and} \\ out_k^{(3)} &= f(\sum_h out_j^{(2)} w_{jk}^{(1)}). \end{cases} \tag{3.4.3}$$

Por lo tanto, se busca el valor más bajo posible de neuronas y capas, pero lo suficiente como para adquirir las características del fenómeno estudiado, sin caer en un sobreajuste (en el que la red neuronal no puede aprender correctamente o generalizarse). Para hacer frente a esto, se considera la siguiente regla:

$$h_1 = (o * r^2), \tag{3.4.4}$$

$$h_2 = (o * r), \tag{3.4.5}$$

$$r = (i/o)^{\frac{1}{3}}, \tag{3.4.6}$$

donde i denota los atributos de entrada, o son las variables de salida, h_1 y h_2 representan el número mínimo para las capas ocultas número 1 y 2, respectivamente, y r es un factor piramidal. Si la neurona no puede aprender en la configuración inicial, el número de neuronas aumenta hasta que se alcanza un rendimiento adecuado [69].

3.4.3. Métricas de evaluación

Con el fin de determinar al algoritmo adecuado de clasificación, se puede elegir el conjunto de casos para inducir al clasificador. En este sentido, se puede usar el método `Holdout`, que particiona el conjunto de datos en dos: entrenamiento y prueba. El grupo de entrenamiento se usa para entrenar al modelo y el de prueba para estimar la tasa de error. El método de `remuestreo`, viene a ser una generalización del método `Holdout`, ya que se realiza este proceso múltiples veces sobre diferentes muestras. Con ello, la tasa de error se calcula a partir de la media de experimentos realizados.

Por otro lado, el método de `Validación cruzada` viene a ser una generalización del método `Holdout`, se basa en la partición de la muestra en K conjuntos de aproximadamente el mismo tamaño, donde $K-1$ constituyen al grupo de entrenamiento y el resto del grupo de prueba. Con ello, se generan $K-1$ modelos para encontrar la tasa media del error [70].

No obstante, el uso de lo métodos antes comentados no son la única forma para una búsqueda adecuada del modelo. Existen 4 parámetros implícitos:

- El punto de inicio de búsqueda: Se basa en determinar la forma de seleccionar el algoritmo de clasificación. Esto puede ser desde el modelo más simple hacia el complejo, viceversa o aleatorio.
- Organización de la búsqueda: La forma en emplear los algoritmos de características de los datos en relación al algoritmo de clasificación.
- La función de evaluación: Viene a ser la puntuación o *score* que mida la calidad de solución del algoritmo, que puede contar diferentes criterios y ponderaciones como rendimiento, precisión, sensibilidad, especificidad, error, entre otros [56].
- Criterio de búsqueda: En consideración con los recursos del sistema embebido, puede ser una restricción en la seleccción del algoritmo de clasificación

Parte IV

RESUMEN DE CONTRIBUCIONES

CAPÍTULO 4: RESUMEN DE CONTRIBUCIONES

El presente capítulo, muestra los resultados obtenidos por medio de las publicaciones de alto impacto. Para ello, en la sección 4.1, se presenta la propuesta realizada sobre los nuevos tipos y niveles de sistemas embebidos enfocados a las tareas de adquisición de datos y la implementación de los algoritmos de aprendizaje automático. Por su parte, la sección 4.2 muestra el esquema de acondicionamiento de datos provenientes de sensores y los resultados obtenidos sobre las técnicas de representación de datos orientadas a su implementación en sistemas embebidos. Con esto, se evidencia el cumplimiento de los objetivos específicos propuestos. Por su parte, la sección 4.3, presenta las conclusiones obtenidas a lo largo del desarrollo de esta tesis. Finalmente, las secciones 4.5, 4.6, 4.7 y 4.8, 4.9 y 4.10 muestran los resúmenes de cada contribución.

4.1. Propuesta

Existe un amplio enfoque de aplicaciones en sistemas embebidos, donde el proceso de adquisición de datos es muy diverso. Sin embargo, su resultado y rendimiento es estrechamente ligado a su capacidad computacional. En este sentido, se presenta una nueva concepción de niveles y tipos de sistemas embebidos en relación a sus capacidades de adquisición y procesamiento de datos (4.1.1 y 4.1.2). Posteriormente, se realiza un nuevo enfoque de taxonomía de sensores relacionados con los temas antes expuestos (4.1.3).

4.1.1. Niveles en sistemas embebidos

La propuesta de niveles en sistemas embebidos se basa en la capacidad de procesar datos e implementar algoritmos complejos. Esto se fundamenta en los cuatro parámetros de desempeño en algoritmos de clasificación supervisada (complejidad de cómputo, tamaño de datos para generación del modelo, tiempo de respuesta y la función de evaluación de cada parámetro) [38]. Debido a esto, se establecen requisitos mínimos en los sistemas embebidos para implementar la capacidad de toma de decisión. Además, al evaluar cada uno de los algoritmos, se generan recomendaciones de usabilidad en base a las condiciones presentadas en el entorno. En consecuencia, en la Fig. 4.1 se muestra los niveles propuestos.

4.1.1.1. Nivel I

Los sistemas embebidos de `Nivel I`, cuentan con las capacidades computacionales de un microcontolador estándar, donde la memoria de almacenamiento (*kilobytes*) del código y sus registros de trabajo son reducidos. En este sentido, la implementación de algoritmos de aprendizaje automático es limitada. En consecuencia, no se cuenta con librerías para este proceso. No obstante, estos sistemas son muy utilizados para la comunicación con sensores, ya que están destinados principalmente a cumplir la función de adquisición de datos. Se vuelven sistemas flexibles que permiten el

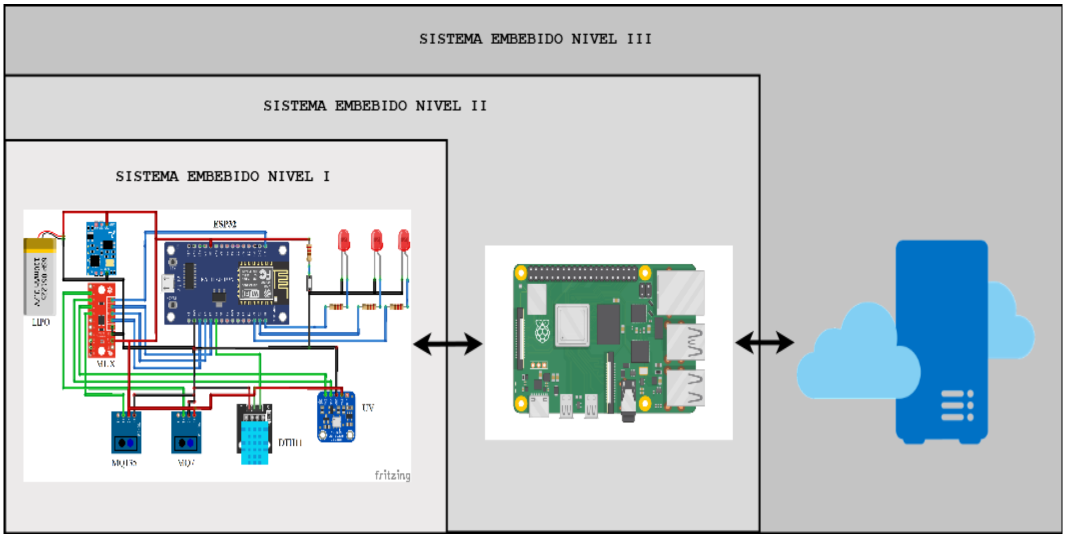


Figura 4.1: Propuesta de niveles en sistemas embebidos

aumento de componentes de *hardware* con facilidad [39].

4.1.1.2. Nivel II

El `Nivel II`, es considerado como un ordenador de placa reducida por su significativo aumento de recursos computacionales, ya que la ejecución de procesos se lo realiza con una memoria dinámica dedicada. Generalmente, se encuentra conectado directamente a un sistema de `Nivel I` por cable o red inalámbrica, ya que su capacidad de interacción con sensores es menor. No obstante, para aplicaciones que requieren visión artificial embebida, son imprescindibles. Esto se debe a que cuentan con *hardware* específico para realizar esta función. Sin embargo, es limitado su uso en aplicaciones de aprendizaje profundo (redes neuronales) [39].

4.1.1.3. Nivel III

Los sistemas de `Nivel III`, tienen mayor capacidad de recursos computacionales por tener la posibilidad de comunicarse con un servidor físico o en la nube. Con ello, los datos no son procesados de forma local. En este sentido, necesitan protocolos livianos de envío de datos que son propuestos bajo estándares de IoT. Además, se pueden integrar sistemas de `Niveles I y II` con facilidad, ya que manejan los mismos protocolos de transferencia de datos [43].

4.1.2. Tipos de sistemas embebidos

Dentro de un sistema embebido de `nivel I`. Existen diferentes características en su diseño acorde a la aplicación específica que debe cumplir. Por tal motivo, se proponen diferentes tipos de sistemas embebidos que son mostrados en la Fig. 4.2 [43].

4.1.2.1. Tipo I

Los sistemas embebidos de `Tipo I`, se consideran de propósito general, ya que cuentan con elementos comunes como periféricos entrada-salida, memorias ROM y RAM, contadores, conversores análogos-digitales y digitales-análogos, interrupciones de estados, entre otros. Es por ello que son muy aptos para conectar sensores de diferentes tipos. Sin embargo, son sistemas que al cumplir con funciones específicas, no tienen la capacidad de enviar información directamente hacia el usuario.

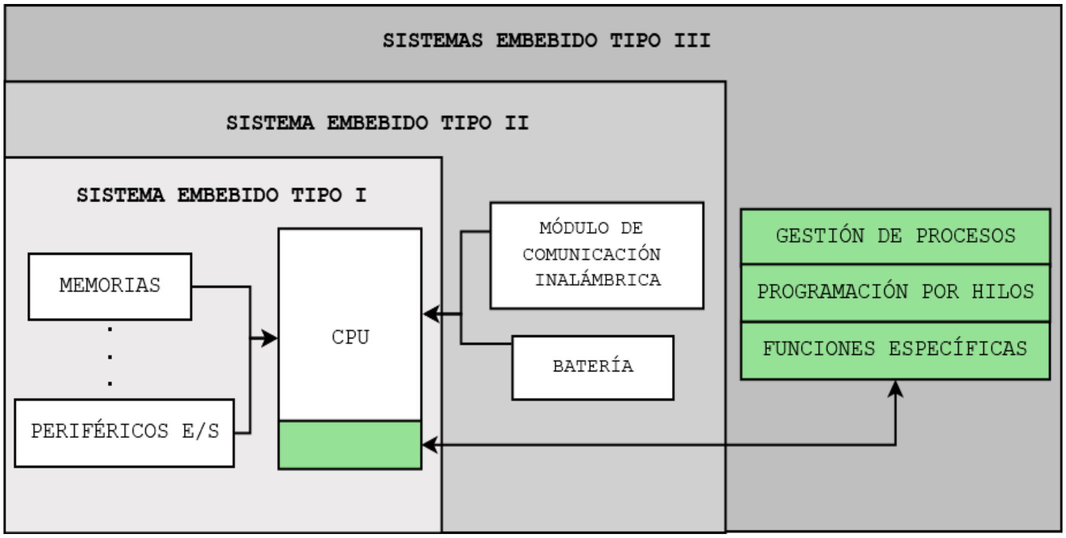


Figura 4.2: Propuesta de tipos de sistemas embebidos

En consecuencia, solo cuentan con módulos de comunicación cableada como: serial, *Inter-Integrated Circuit* (I2C), entre otras [42].

4.1.2.2. Tipo II

Los sistemas considerados de `Tipo II`, cuentan con un módulo de comunicación inalámbrica que les permite realizar el envío de datos de forma remota. Con ello, según el protocolo de comunicación que sea implementado, pueden compartir información entre otros nodos similares al definir una topología de red. Este concepto es muy utilizado en las WSN y textiles inteligentes. Sin embargo, la gestión de la batería sigue siendo una problemática abierta debido a que son sistemas portables ubicados en sectores de difícil acceso.

4.1.2.3. Tipo III

Los sistemas embebidos de `Tipo III`, se enfocan principalmente en la gestión de los recursos computacionales de un microcontrolador. En consecuencia, pueden tener módulos extras que les permitan procesar información de forma específica. Como principales ejemplos, se cuenta por un lado, los sistemas RTC, por su facilidad de implementar código hacia el propio compilador del sistema. Por otro lado, los sistemas DSP, son diseñados con una estructura más avanzada que permite manejar vectores de gran dimensión para procesar diferentes tipos de señales [40].

4.1.3. Taxonomía de sensores

Con el avance tecnológico, los sensores han ampliado su rango de aplicación, a tal punto que se pueden encontrar en cualquier rama del conocimiento. En muchos casos, para mejorar la interacción entre los sensores y el sistema embebido, los desarrolladores proporcionan sistemas de acoplamiento con el fin de mejorar el proceso de envío de datos. Sin embargo, en relación al fenómeno estudiado y las condiciones reales de funcionamiento, es necesario contar con sistemas robustos de filtrado de datos y suavizado de la señal [44]. Este proceso está ligado a la forma de envío de los datos desde el sensor hacia el sistema que procesa la señal. Por esta razón, se propone una taxonomía de sensores que los clasifique según este criterio. Esto se observa en la Fig. 4.3.

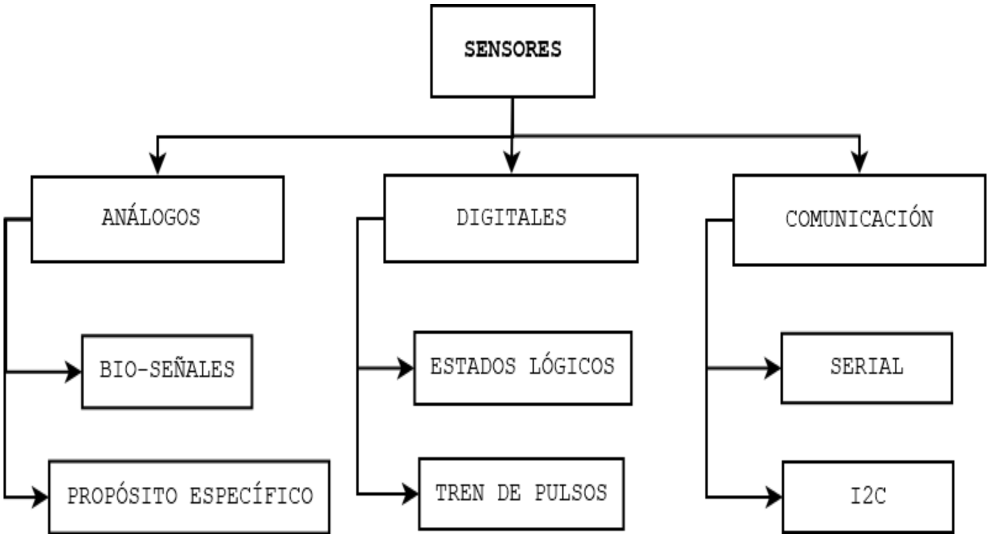


Figura 4.3: Taxonomía propuesta de sensores

4.1.3.1. Sensores análogos

La principal forma de adquirir datos de un fenómeno estudiado es mediante la variación del voltaje del transductor al recibir las cambios de la señal de entrada. Estos tipos de sensores, en relación a su aplicación se dividen en dos grupos. Por un lado, se encuentran los sensores de bio-señales, ya que cuentan con un filtrado muy similar al recopilar información del cuerpo humano. Esto se debe a que estas señales se encuentran en bajas frecuencias. Por otro lado, los de propósito específico no cuentan con una frecuencia estándar de funcionamiento. Por esta razón, su filtrado se enfoca en la no-linealidad de los elementos electrónicos y se orienta a usar técnicas de suavizado de la señal en *software* [49].

4.1.3.2. Sensores Digitales

En algunos casos, los sensores solo pueden detectar dos estados (nivel alto y bajo). Esta información es simple de adquirir y procesar. Sin embargo, en un lenguaje digital (0 a 5 voltios), existe un rango de incertidumbre entre 1.3 a 2.7 voltios que el sistema no los reconoce adecuadamente y genera errores de lectura.
Existen otros tipos de sensores digitales que hacen uso de un tren de pulsos. Su funcionalidad se basa en una comparativa interna para expresar un valor del fenómeno estudiado. En estos casos, usualmente se necesita de librerías propias del fabricante para la interpretación correcta de los datos obtenidos.

4.1.3.3. Sensores por comunicación

En la actualidad, se busca simplificar el proceso de diseño de circuitos electrónicos y realizar sistemas modulares (fácil integración entre sistemas). Con ello, los sistemas se vuelven flexibles ya que pueden modificar su funcionalidad de ágil manera. Esto ayuda a un rápido prototipado, ya que cuentan todos los elementos electrónicos con el protocolo de comunicación I2C. Con esto, la identificación de cada sensor es mediante un código hexadecimal. Cabe recalcar que cada sensor con este estándar de comunicación cuenta con librerías muy depuradas de adquisición de datos [47].

4.1.4. Análisis de datos

Los criterios de análisis de datos establecidos sobre los algoritmos de aprendizaje supervisado pueden ser empleados para mejorar el conjunto de entrenamiento al modelar un algoritmo de clasificación, este proceso influye directamente en la selección del tipo y nivel de sistema embebido. En consecuencia, se plantea una metodología de análisis de datos que cuenta con métricas de rendimiento ligadas al coste computacional. Con ello, se busca la selección adecuada de criterios y algoritmos que permitan mantener un alto rendimiento de clasificación y que no afecte al tiempo de respuesta del tipo de sistema embebido empleado. Es importante recalcar, que este proceso es llevado a cabo con el fin de contar con un sistema adaptable que pueda modificar sus criterios (conjunto de entrenamiento) sin la necesidad de emplear elementos computacionales externos como computadores o servidores.

En este sentido, se emplean las bases de datos multivariantes obtenidas por los sistemas embebidos desarrollados aplicando el acoplamiento y filtrado de *hardware y software* necesario. Cada una de ellas se han dividido aleatoriamente en 10 conjuntos de muestras para obtener los datos de entrenamiento y de prueba, asignando un porcentaje de instancias del 80 % y 20 % respectivamente. Además, cada algoritmo es usado en su configuración estándar para evitar sesgos de experiencias de uso sobre alguno de ellos. En la Fig. 4.4 se muestra, la metodología propuesta, a fin de determinar los criterios y algoritmos adecuados para cada posible aplicación en los sistemas embebidos.

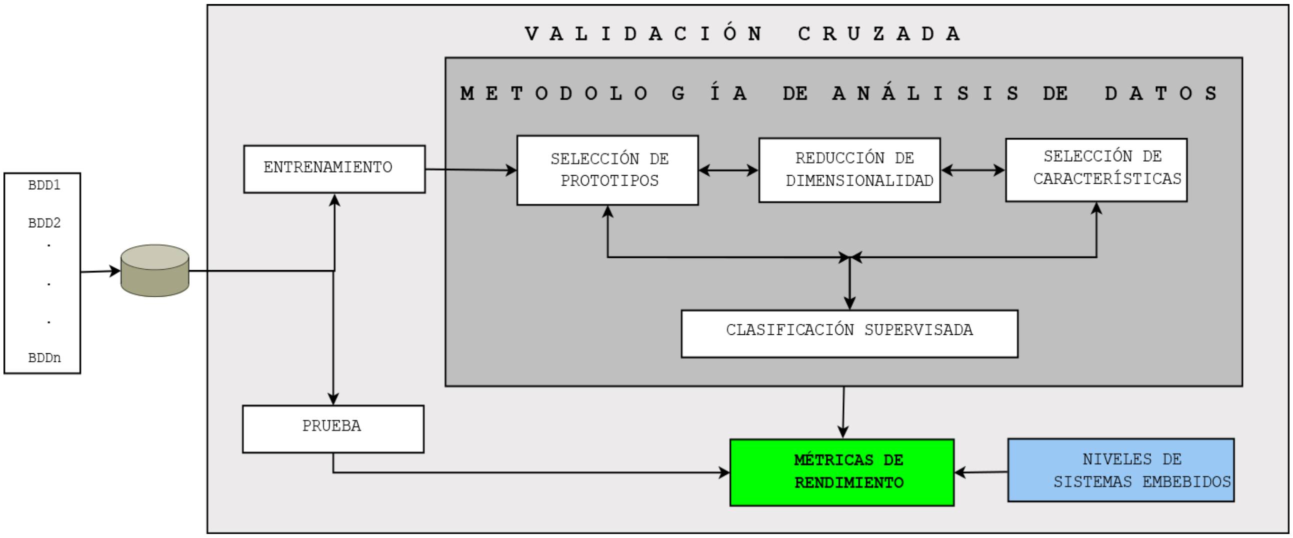

VALIDACIÓN CRUZADA
METODOLOGÍA DE ANÁLISIS DE DATOS
BDD1
BDD2
.
.
.
BDDn
ENTRENAMIENTO
SELECCIÓN DE PROTOTIPOS
REDUCCIÓN DE DIMENSIONALIDAD
SELECCIÓN DE CARACTERÍSTICAS
CLASIFICACIÓN SUPERVISADA
PRUEBA
MÉTRICAS DE RENDIMIENTO
NIVELES DE SISTEMAS EMBEBIDOS

Figura 4.4: Metodología de análisis de datos

4.2. Resultados

Los resultados presentados hacen referencia a los objetivos específicos cumplidos en esta tesis doctoral. En este sentido, se enfocan en dos componentes principales. El primero, es la parte electrónica sobre el acomplamiento y acondicionamiento de datos (sección 4.2.1). Segundo, la presentación de datos y el enfoque multi-criterio de optimización en el coste computacional para la implementación de algoritmos de aprendizaje automático en sistemas embebidos (sección 4.2.2). Cabe recalcar, que este proceso se realizó en un computador estándar con un procesador INTEL CORE I7 con 16 GB en MEMORIA RAM y sistema operativo WINDOWS 10. Con ello, solo los algoritmos adecuados se diseñarán hacia una infraestructura 8, 16 o 32 bits de un sistema embebido.

4.2.1. Esquema de acondicionamiento de datos

Se han evaluado los sensores empleados en las publicaciones realizadas con métricas de rendimiento tomando como referencia sistemas robustos de cada área del conocimiento. Con ello, se define la capacidad de cada sensor para que pueda utilizarse en condiciones reales. Además, se pueden establecer parámetros y criterios de diseño en el desarrollo de sistemas embebidos y las diferentes precauciones a tomar en cuenta para evitar el desgaste y pérdida de calibración de cada sensor. Esto se muestra en la tabla 4.1. Los resultados obtenidos del voltaje de los sensores se comprobaron con un equipo de mayor prestación y sensiblidad a ligeras variaciones de la señal de entrada que un sistema embebido, como es el multímetro `KEYSIGHT DIGITAL MULTIMENTER U1282A`, este dispotivo tiene un 0,025 % de precisión de voltaje. Por otro lado, para mayor entendimiento del resultado de cada métrica, se establecen 4 niveles de valoración hacia los sensores: (i) Excelente, (ii) Bueno, (iii), Normal y (iv) Deficiente.

Cuadro 4.1: Resumen de tipos de sensores en relación a sus métricas de rendimiento

Métricas	**Tipos de Sensores**				
	Bioseñales	**Propósito específico**	**Estados Lógicos**	**Tren de pulsos**	**Comunicaciones**
Exactitud	Bueno	Bueno	Excelente	Excelente	Excelente
Precisión	Bueno	Normal	Excelente	Bueno	Excelente
Reproducibilidad	Normal	Normal	Deficiente	Deficiente	Bueno
Repitibilidad	Bueno	Deficiente	Excelente	Bueno	Bueno
Estabilidad	Excelente	Bueno	Bueno	Deficiente	Bueno
Ruido	Bueno	Bueno	Bueno	Bueno	Bueno

Los experimentos de adquisición de datos con cada uno de los sensores se realizaron de forma similar en 10 ocasiones. Esto permite contar con una estabilidad del proceso realizado. Además, en relación a coste computacional y tiempo de respuesta del sistema, se definen los algoritmos de suavizado de la señal en relación a la taxonomía de sensores y el tipo de sistema embebido. Esto se aprecia en la tabla 4.2.

Cuadro 4.2: Elección de técnicas de filtrado y suavizado de la señal en sensores

Tipo de sensor	Tipo de Sistema embebido	
	`TIPO I/II`	`TIPO III`
Bioseñales	IIR Nuttall	FIR Chebyshev
Propósito específico	Guassiano	Savi-Golay
Estados Lógicos		
Tren de pulsos	Guassiano	Savi-Golay
Comunicaciones	Mediano	Savi-Golay

4.2.2. Técnicas de representación de datos

El criterio de selección de prototipos busca eliminar la redundancia y errores en los datos con el fin de encontrar la mejor base de datos posible. Esto ocasiona una reducción significativa en su volumen y permite la flexibilidad de compilar algoritmos complejos dentro de un sistema embebido. En consecuencia, se debe utilizar una métrica que permita seleccionar la combinación adecuada entre el número de instancias removidas (IR), el rendimiento de clasificador (RC) y el tiempo de respuesta del sistema (TR). Como resultado, la métrica IRT (Instancias removidas, Rendimiento del clasificador y Tiempo de respuesta) es mostrada en la ecuación 4.2.1.

$$\text{IRT} = \frac{(IR * RC)}{TR} * 100\,\% \tag{4.2.1}$$

Como resultado de este análisis, se presenta un diagrama de flujo de selección de algoritmos con respecto al nivel de sistema embebido empleado para una determinada aplicación (Fig. 4.5).

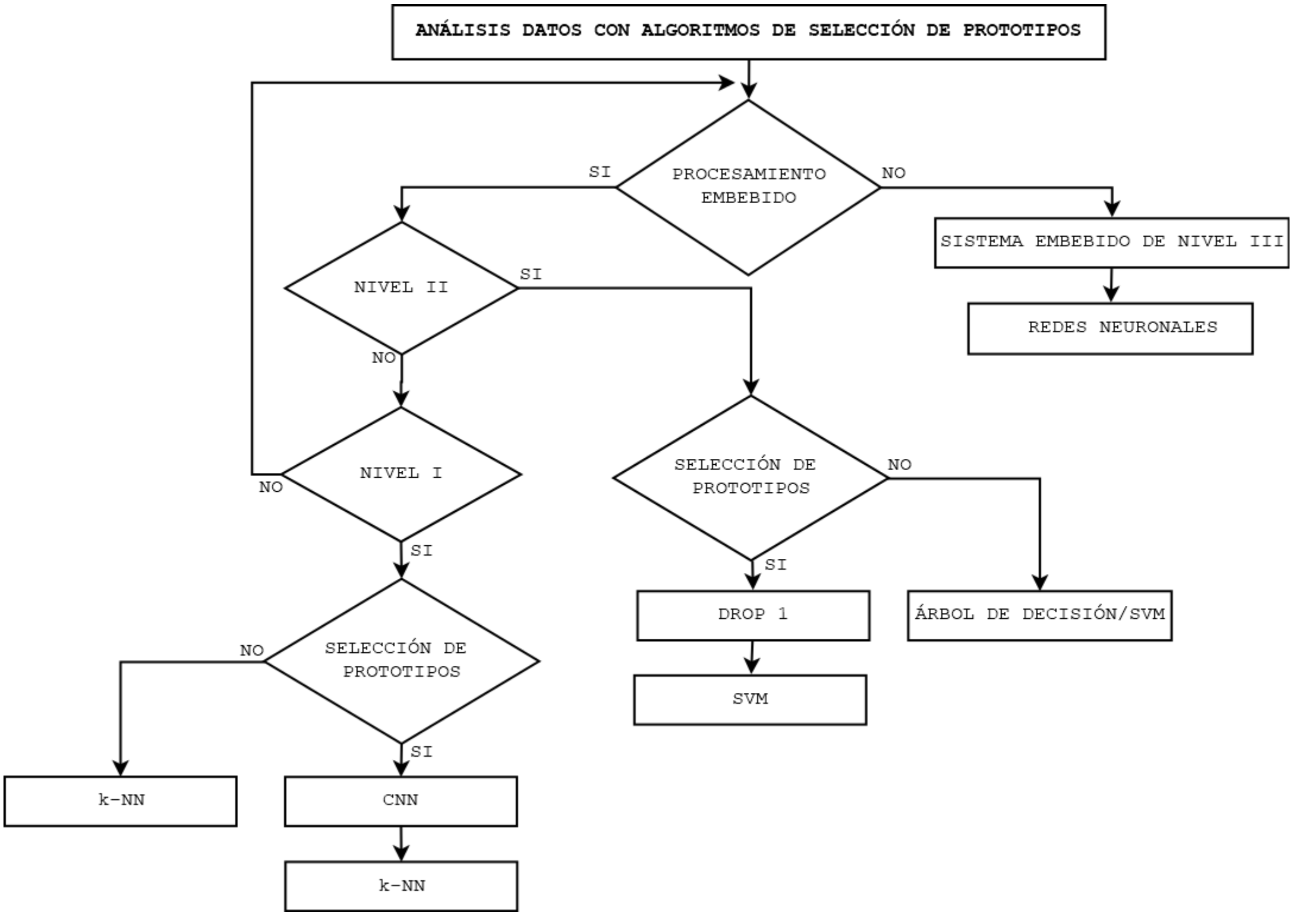


Figura 4.5: Diagrama de flujo sobre la implementación de algoritmos de selección de prototipos en sistemas embebidos

La reducción de dimensionalidad es empleada generalmente con el fin de mostrar la información en una dimensión capaz de ser entendible para el ser humano. En este caso en dos o tres dimensiones. En este sentido, los algoritmos de reducción de dimensionalidad han sido usados con el fin de contar una base de datos de entrenamiento de dos dimensiones. En consecuencia, se logra una significativa reducción en el proceso de almacenamiento y procesamiento de datos. No obstante, algunos algoritmos de RD, no puedieron mantener el rendimiento de clasificación y son descartados para su uso.

En relación al coste computacional y la facilidad de implementar los criterios de reducción de dimensionalidad, se presenta un diagrama de flujo acorde a la aplicación de este criterio conforme el nivel de sistema embebido. Esto se muestra en la Fig. 4.6.

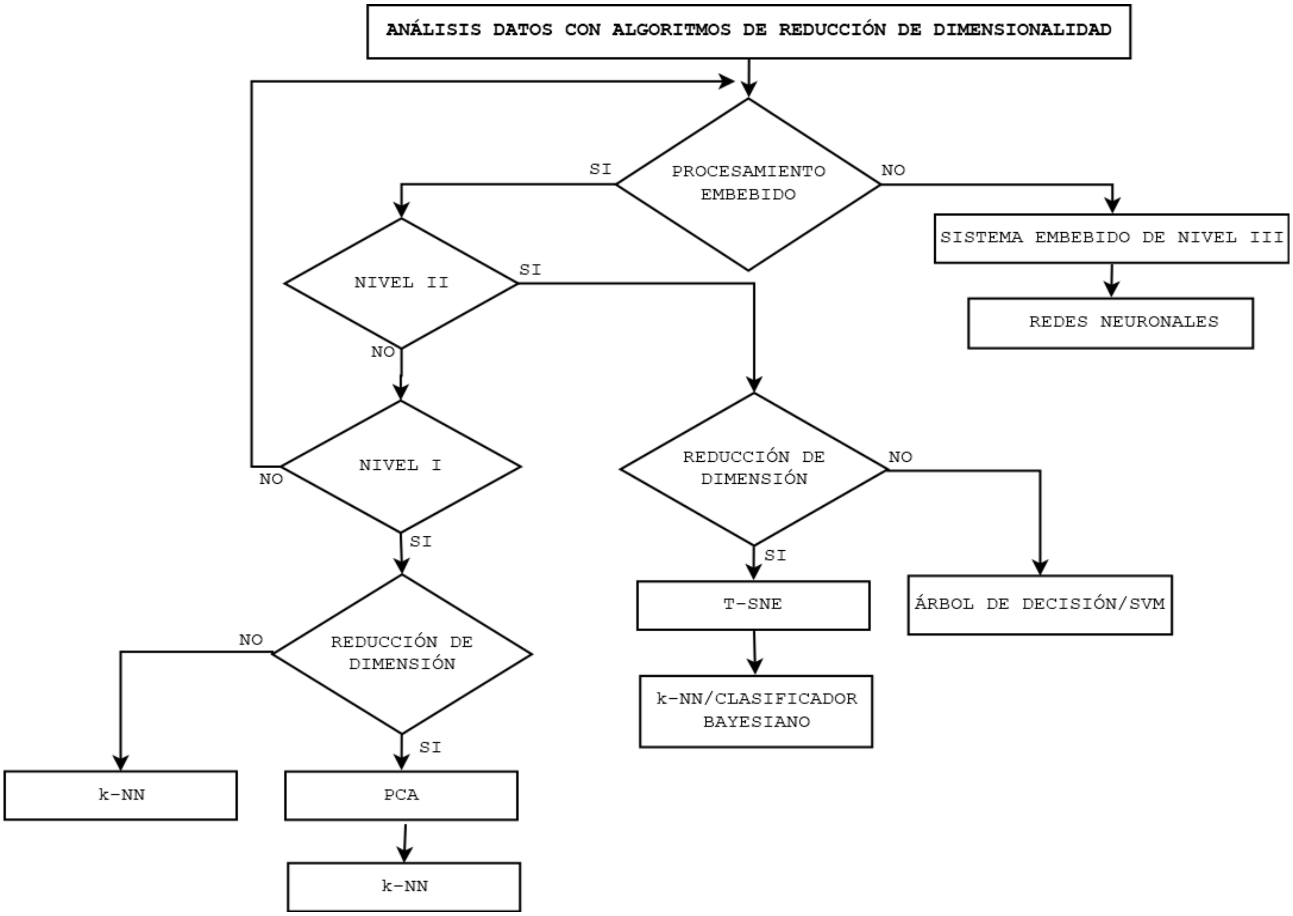


Figura 4.6: Diagrama de flujo sobre la implementación de algoritmos de reducción de dimensionalidad en sistemas embebidos

4.3. Conclusiones

- La clasificación por niveles y tipos en sistemas embebidos es de gran utilidad en el momento de definir la aplicación y consumo computacional que necesita representar un fenómeno estudiado. Con ello, la selección de elementos electrónicos activos y pasivos se vuelve una tarea eficiente y con mejores resultados.

- Existen una gran cantidad de taxonomías acerca del diseño y construcción de los sensores. No obstante, este nuevo enfoque presentado se encuentra relacionado a la interacción con sistemas embebidos y su capacidad de calibrar y mejorar la señal proveniente de sensores. Por tal motivo, es de gran utilidad y brinda la oportunidad de mejorar la robustez en la toma de decisión de los algoritmos de aprendizaje de automático.

- La selección de prototipos es un criterio adecuado en reducir el tamaño de la matriz de entrenamiento. En consecuencia, el conocimiento intrínseco de los datos puede verse afectado en cierta medida. Sin embargo, los algoritmos `CNN` y `DROP1` demostraron ser los adecuados acorde a la capacidad computacional del sistema embebido a desarrollar.

- Los algoritmos de reducción de dimensionalidad permiten agrupar la información de los atributos establecidos en conjuntos de datos de menor dimensión. Como se pudo comprobar, `PCA` y `t-SNE` son los algoritmos adecuados para su implementación.

- La metodología propuesta de análisis de datos permitió la selección de los algoritmos según los criterios de características de los datos. No obstante, no se realizaron pruebas de funcionamiento entre ellos. Sin embargo, se propone, usar varios de estos criterios, comenzar con una etapa de selección de características, luego, la reducción de dimensionalidad y finalmente, la selección de prototipos.

4.4. Trabajos Futuros

Como trabajos futuros, se plantea seguir realizando sistemas embebidos en diferentes áreas del conocimiento, aplicando la metodología propuesta y profundizando en la utilización de sistemas híbridos (sensores y visión por computador). Por este motivo, se debe indagar a fondo en la implementación de algoritmos de aprendizaje profundo aplicados a la gestión de procesos en tiempo real.

Por otra parte, se considera necesario contar con un sistema robusto de pruebas de sensores que permita emular las condiciones reales de funcionamiento. Con esto, las métricas de rendimiento de cada sensor, se obtendrán mediante reportes especializados.

4.5. Contribución 1

4.5.1. Título

P. D. Rosero-Montalvo, D. H. Peluffo-Ordóñez, V. F. López Batista, J. Serrano and E. A. Rosero, “Intelligent System for Identification of Wheelchair User’s Posture Using Machine Learning Techniques,” in IEEE Sensors Journal, vol. 19, no. 5, pp. 1936-1942, 1 March1, 2019.

4.5.2. Objetivos

Las personas que usan sillas de ruedas pueden tener problemas, físicos, mentales y/o discapacidad sensorial que limita sus actividades diarias. Alrededor del mundo, aproximadamente el 15 % de la población tiene algún tipo de problema de movilidad. Aunque se ha demostrado que el uso de silla de ruedas aumenta la calidad de vida de los usuarios al permitir la movilidad, ocupación e interacción social, entre otros. Su estado de salud puede ser muy afectado por una mala postura. Ya que esto puede provocar dolor crónico, esclerosis, cifosis, problemas cutáneos y respiratorios, pérdida de habilidades cerebrales, algunos problemas de salud física como rigidez muscular, fatiga y dolor muscular, entre otros. Las causas pueden ser por la falta y desequilibrio muscular, mala posición del cojín del asiento de la silla de ruedas y la fatiga al estar en una sola posición. En algunos casos, el usuario carece de la sensación de dolor, hasta que su salud se haya visto seriamente comprometida. Por otro lado, se logran múltiples beneficios cuando el usuario muestra el hábito de sentarse correctamente: la intensidad del dolor disminuye y la probabilidad de úlceras se reduce significativamente.

El objetivo principal de este trabajo es realizar un sistema electrónico con un arreglo de sensores integrados en el cojín del asiento y del respaldo. Los datos recopilados por los sensores se utilizan para la detección de postura de la persona mediante el uso de algoritmos de clasificación supervizada. En este sentido, se emplean técnicas de balanceo de datos, selección de prototipos y reducción de dimensionalidad para encontrar una base de datos con el menor tamaño posible. Para probar el sistema se diseñan experimentos controlados para la adquisición de datos de las 4 posiciones más comunes en una silla de ruedas, donde 3 de ellas son erróneas.

4.5.3. Metodología

La metodología planteada parte con un adecuado diseño del sistema electrónico al establecer los requisitos de usuario. Con ello, se determina la posición de los sensores de presión en el asiento y un sensor que mide la distancia en el respaldo. Posteriormente, se realiza la toma de datos con varios usuarios que recrean las 4 posiciones planteadas y que son almacenadas con su correspondiente etiqueta. Finalmente, se emplea un esquema de análisis de datos con las etapas de: (i) la selección de prototipos para la eliminación de datos erróneos. (ii) el criterio de balanceo de datos para tener una base de datos con el mismo valor de instancias por etiqueta. (iii) la reducción dimensionalidad de la base de datos a dos dimensiones, para almacenar este conjunto de datos resultante dentro del sistema embebido y, (iv) se codifica el algoritmo *K-Nearest Neighbors* (k-NN) como algoritmo de clasificación.

4.5.4. Resultados

El sistema propuesto tuvo una reducción de la base de datos para la implementación del algoritmo de aprendizaje automático del 88 %. De esta forma, el microcontrolador pudo relizar el proceso de clasificación en un tiempo óptimo (menos de un minuto) sin sobrecargar sus recursos computacionales. En condiciones reales, el sistema alcanza una precisión de clasificación del 75 %. Cabe recalcar, que al ser implementado el sistema, brinda una alarma en forma de vibraciones en la propia silla de ruedas para la corrección de postura por parte del paciente.

4.5.5. Conclusiones

En este trabajo, se ha propuesto el diseño de un sistema embebido que incorpora un algoritmo de clasificación dentro la memoria dinámica del microcontrolador que, con el empleo de las etapas de reducción del tamaño del conjunto de entrenamiento, es capaz de detectar defectos de la postura de una persona sentada en una silla de ruedas. Por lo tanto, esta representación mínima de datos permite a nuestro sistema almacenar y clasificar posturas en tiempo real.

El rendimiento del sistema se consideró aceptable dados sus recursos computacionales y condiciones de contexto. Una característica importante a destacar es que el sistema responde a una mala postura en aproximadamente un segundo, lo que permite la generación de un aviso adecuado al usuario.

4.6. Contribución 2

4.6.1. Título

Paul D. Rosero-Montalvo, Vivian F. López Batista, Jaime Riascos, Diego Hernán Peluffo-Ordóñez: "Intelligent WSN System for Water Quality Analysis Using Machine Learning Algorithms: A Case Study (Tahuando River from Ecuador)". Remote. Sens. 12(12): 1988 (2020).

4.6.2. Objetivos

Los ríos son corrientes naturales de agua que desembocan en lagunas o el mar. En las ciudades, son los encargados de brindar este recurso hídrico a la agricultura, la industria, entre otros. Sin embargo, este recurso natural se ha vuelto escaso. Esto se debe a la deforestación, su uso inadecuado y el excesivo empleo de fertilizantes y pesticidas que han causado problemas ambientales. Asi mismo, la creciente urbanización y aumento de industrias han impactado severamente en la calidad del agua en ecosistemas de ríos a nivel mundial. Además, con el aumento de la población, las aguas residuales ingresan a los ríos sin ningún control ambiental. Las Naciones Unidas han determinado que el 90 % de estos residuos no son tratados y el 70 % de la industria descarga contenido contaminante sin estándares adecuados ni inspecciones rigurosas.

El objetivo de esta publicación es determinar el nivel de contaminación en el río Tahuando de la ciudad Ibarra-Ecuador. Para hacer esto, se realiza la toma de datos en diferentes sectores del recorrido del río donde se evidencia el impacto de la contaminación. En consecuencia, se calibran y acondicionan los sensores bajo diferentes métricas de rendimiento en relación a equipos especializados. Con ello, se realiza una etapa de análisis de datos para implementar los criterios de clasificación en una red de sensores inalámbricas que monitorean el río y generan una semaforización del estado del mismo.

4.6.3. Metodología

La metodología empleada empieza con el acondicionamiento de los sensores y su rendimiento en relación a los aparatos de medida robustos que se utilizan para medir la turbidez, pH, cantidad de óxido disuelto y temperatura del agua. Después de la toma de muestras, se realizaron pruebas de confiabilidad, precisión, disponibilidad, facilidad de uso y escalabilidad. Posteriormente se empleó una red de sensores inalámbricos para el envío de datos a un servidor externo. Por otro lado, con la base de datos obtenida, se diseña un esquema de análisis de datos con las etapas: (i) selección de prototipos y (ii) comparación de los diferentes criterios de clasificación para seleccionar el adecuado de cara a su implemetación en cada nodo WSN.

4.6.4. Resultados

Los resultados relacionados al desarrollo de la WSN se enfocaron en la utilización de los algoritmos de suavizado de la señal. El filtro promedio demostró ser el adecuado con respecto al tipo de datos provenientes de los sensores. Por su parte, el algoritmo *Condensed Nearest Neighbor* (CNN), correspondiente a la etapa de la selección de prototipos pudo eliminar en un 97 % la base de entrenamiento. Al usar el criterio de clasificación basado en distancias como lo es el algoritmo *K-Nearest Neighbors* (k-NN), se tuvo un rendimiento del 90 %. Finalmente, se empleó una métrica de balance cualitativo entre las instancias eliminadas y el rendimiento de clasificación que se denominó *Quantitative Metric of Balance* (QMB).

4.6.5. Conclusiones

Para el diseño electrónico, dado que el río del caso de estudio puede tener altos niveles de contaminación, así como también pueden ocurrir variaciones significativas en función de las horas del día y zonas de su recorrido, se implementaron varios nodos WSN para adquirir las condiciones del río. Cubriendo una zona significativa y dentro de un intervalo de tiempo suficientemente amplio. En este sentido, se calibraron los sensores para una correcta adquisición de datos. Además, se demostró experimentalmente que nuestros horarios de lectura de datos eran adecuados para detectar horas de mayor contaminación. Además, destacamos que las boyas fluviales son un elemento clave para cumplir con los requisitos de permeabilidad del nodo, así como para permitir el correcto funcionamiento de cada nodo WSN. Con respecto al esquema de análisis de datos propuesto, se demostró que el uso de un clasificador junto a una buena selección de prototipos es adecuado para un sistema de monitoreo de la calidad del agua basado en WSN.

4.7. Contribución 3

4.7.1. Título

Paul D. Rosero-Montalvo, Vanessa Erazo-Chamorro, Vivian F. López-Batista, María Moreno-García, Diego H. Peluffo-Ordóñez: " Environment Monitoring of Rose Crops Greenhouse Based on Autonomous Vehicles with a WSN and Data Analysis".

4.7.2. Objetivos

El cultivo de rosas tiene un gran impacto en la economía de Ecuador, ya que estas flores se exportan y abarcan el 9 % del mercado mundial. El cultivo de rosas aporta aproximadamente 500 millones de dólares al presupuesto nacional y cubren 8.000 hectáreas en el país. Con la creciente demanda de cultivo de flores, un entorno natural no siempre es el óptimo para lograr los requisitos necesarios de crecimiento de rosas. Condiciones extremas como la exposición directa al sol, el granizo, las enfermedades y las plagas pueden afectar seriamente la calidad del producto y el volumen de producción. Por esta razón, los invernaderos de gran tamaño son cada vez más populares, ya que pueden modificar las condiciones ambientales del interior mediante luces, ventilación, calefacción, entre otros. Por lo tanto, los ciclos de producción de cultivos se pueden planificar en función de las necesidades del mercado.

El sistema propuesto está compuesto por nodos WSN implementados en vehículos autónomos cuadrúpedos que se mueven dentro de un invernadero. En primer lugar, cada nodo WSN tiene un conjunto de sensores que monitorean la humedad relativa, CO_2, temperatura ambiente, cantidad de luz y humedad del suelo. Esto se hace de tal manera que todo el proceso de análisis de datos se puede incorporar y ejecutar dentro de cada nodo WSN con un bajo consumo de recursos computacionales. En segundo lugar, el vehículo cuadrúpedo está diseñado para evitar colisiones y moverse en el invernadero a través del sistema de posicionamiento global (GPS).

4.7.3. Metodología

La metodología de este trabajo cuenta con dos fases. La primera comienza con el adecuado proceso de adquisición de datos por parte de los sensores con métricas de rendimiento de funcionalidad y el movimiento del vehículo autónomo al aplicar marcas con coordenadas GPS dentro del invernadero para establecer giros y lugares de tomas de datos. Como segunda, se propone un esquema de análisis de datos con las etapas de (i) balanceo de datos, (ii) selección de prototipo y (iii) clasificación supervisada para determinar los algoritmos que mejor se adapten al tipo de datos provenientes de los sensores.

4.7.4. Resultados

Los nodos configurados de la WSN (tres en total), se colocaron dentro de una carcasa de material aglomerado, por su fácil corte, diseño y durabilidad. Una vez configurados y en funcionamiento los nodos WSN, se implementa la conexión hacia los vehículos autónomos. Estos se comunican a través del envío de datos en serie. De esta forma, cuando el vehículo autónomo alcanza su marca, envía un bit de activación al nodo WSN para adquirir los datos provenientes de los sensores. El vehículo se inclina hacia abajo para enterrar el sensor de humedad y envía esos datos con su ubicación GPS al nodo WSN. Luego, este envía un bit de fin de proceso. Posteriormente, el vehículo vuelve a moverse hasta encontrar otra marca GPS.

Con el vehículo autónomo montado y todas sus partes configuradas, se realizaron las pruebas de rendimiento con el fin de establecer dos aspectos. El primero, es la duración de la batería, donde el vehículo funcionó sin modos de ahorro de energía y los nodos WSN con todos los sensores activos. Como resultado, la batería del vehículo autónomo duró aproximadamente 8 horas y la del nodo WSN 6 horas. El segundo aspecto se refiere a los horarios del modo de ahorro de energía en los nodos de la WSN funcionaron normalmente durante 14 horas.

Se evaluó la toma de decisiones del sistema en relación a personas expertas (experiencia práctica en cultivos y equipos de medición ambiental). Estas pruebas se realizaron para definir la acción correcta del sistema en las diferentes ubicaciones del invernadero. Se realizaron cuarenta pruebas de funcionamiento para evaluar la capacidad de decisión del sistema. El sistema tuvo un 97,5 % de éxito en las acciones realizadas dentro del invernadero. Como resultado, las rosas tienen un mayor crecimiento del tallo y una mejor frondosidad en el cultivo. En términos de retorno de la inversión, la implementación de este sistema tuvo un margen inicial de incrementar un 5 % el beneficio neto del cultivo. Esto está relacionado con el menor consumo de agua (15 % por ciento), menor uso de pesticidas (8 % por ciento) y el resultado de la venta de rosas (3 %).

4.7.5. Conclusiones

La WSN con vehículos autónomos cuadrúpedos cumple el objetivo de brindar información sobre el cultivo de rosas por sectores dentro del invernadero. En consecuencia, el esquema planeado para el análisis de datos fue adecuado. Permitió una reducción significativa de datos redundantes y el uso de algoritmos de clasificación computacionalmente livianos que se pueden implementar en nodos WSN con recursos limitados. Además, permite recopilar una gran cantidad de información que puede ser útil en los próximos años, ayudando a los agricultores a modificar sus técnicas de cultivo con respecto al cambio climático.

Con el vehículo autónomo, fue posible organizar adecuadamente los ciclos de crecimiento de las rosas. De esta forma, proponemos un nuevo enfoque en el diseño y construcción de invernaderos, que permite la flexibilidad que necesita el cultivo (ventiladores y compuertas en diferentes ubicaciones, no centralizadas). De esta manera, se puede realizar un análisis más extenso con respecto al cambio de parámetros ambientales con el fin de encontrar valores óptimos de crecimiento y mejorar la calidad del producto.

Este artículo fue el resultado del proyecto financiado por la Fundación General de la Universidad de Salamanca en la CONVOCATORIA PARA LA REALIZACIÓN DE PRUEBAS DE CONCEPTO Y PROTECCIÓN DE RESULTADOS, PLAN TCUE 2018-2020: “Sistema Autónomo de Reconocimiento de Patrones en Invernaderos”.

4.8. Contribución 4

4.8.1. Título

Paul D. Rosero-Montalvo, Vivian F. López-Batista, Ricardo Arciniega-Rocha, Diego H. Peluffo-Ordóñez: “Air Pollution Monitoring Using WSN nodes with Machine Learning Techniques: A case study)”.

4.8.2. Objetivos

Los diferentes microclimas del planeta están fuertemente conectados. Esto se debe a diferentes factores como las corrientes marinas, el clima, el movimiento de la luna, entre otros. Estas variables influyen en la temperatura, la humedad, la presión atmosférica y la precipitación en diferentes continentes. En este sentido, se convierte en un sistema muy complejo y cualquier alteración puede causar un impacto grave en el planeta. En los últimos años, una de las mayores preocupaciones a nivel mundial es el aumento de la temperatura del planeta. Esto produce variaciones climáticas que, por un lado, pueden generar olas de calor excesivas que erosionan el suelo provocando la muerte de animales y plantas. Por otro lado, lluvias agresivas generan inundaciones, desbordes de ríos, entre otros. Esto se debe principalmente al crecimiento descontrolado de las industrias que provoca el exterminio de los bosques y generan contaminación del aire y el agua. Estos efectos de la industrialización, junto con la urbanización y la movilidad individual de las personas, se han convertido en un gran riesgo para la salud. En consecuencia, la Organización Mundial de la Salud (OMS) estima que una de cada ocho muertes prematuras se debe a los efectos de la contaminación del aire. En consecuencia, se puede deducir que alrededor de 3 millones de personas mueren por contaminación atmosférica anualmente.

Por las razones antes mencionadas, el objetivo de este trabajo de investigación es el diseño y desarrollo de una red de sensores inalámbricos (13 nodos en total) para monitorear los gases y contaminantes del aire más comunes como son: óxido de nitrógeno (NO_x) y el monóxido de carbono (CO). Además, se utiliza un sensor de rayos ultravioleta (UV) para determinar las tasas máximas de radiación y el sensor de temperatura y humedad relativa con el fin de conocer su relación con la contaminación del aire. Para lograrlo, se realiza un análisis de datos con el fin de determinar los índices de contaminación dentro de la ciudad observados desde una interfaz en un servidor local.

4.8.3. Metodología

El esquema de análisis de datos propuesto empieza con una etapa comparativa de los diferentes algoritmos de suavizado de la señal que elimine los errores de los datos provenientes de los sensores. Con ello, la siguiente etapa busca reducir la base de datos de entrenamiento mediante los criterios de selección de prototipos y selección de características por medio de una comparativa entre ellos para elegir los algoritmos que mejor se adapten a un criterio de clasificación.

4.8.4. Resultados

Se determina que el filtro Gaussiano es el adecuado para su implementación en todos lo sensores ya que su relación señal/ruido es mejor y su coste de implementación no altera el tiempo de respuesta. El algoritmo *Condensed Nearest Neighbor* (CNN) demuestra tener un mayor porcentaje de eliminación de datos del conjunto de entrenamiento y que el uso de redes neuronales, si se tiene la posibilidad de emplear un servidor externo es adecuado. Además, si se desea integrar la capacidad de decisión en cada nodo de monitoreo, *k-Nearest Neighbors* (k-NN) demotró ser el que emplea la menor cantidad de recursos computacionles. Finalmente, se pudo comprobar que el algoritmo Relief F, permite tener una análisis de relevancia de variables de forma estadística muy acorde para la selección de características para entrenar un modelo de clasificación supervisada. Se logra un rendimiento de clasificación del 95 % con una reducción signicativa de los datos ruidosos.

4.8.5. Conclusiones

La integración de los nodos WSN para el monitoreo de las condiciones de contaminación del aire en la ciudad de Ibarra brindó información importante para clasificar la contaminacion en los niveles: alto, medio y no perceptible. Con esto se puede planificar la implementación de áreas verdes dentro de la ciudad. Además, se pudo validar el correcto funcionamiento del sistema y la forma en que los algoritmos de aprendizaje automático se adaptan a los cambios para la toma de decisiones. De esta manera, los protocolos inalámbricos utilizados (WiFi y 4G) son estables para el envío de datos. La metodología de análisis de datos propuesta, a partir del suavizado de datos, tuvo los criterios correctos para brindar información adecuada al clasificador para el entrenamiento de su modelo.

4.9. Contribución 5

4.9.1. Título

Rosero-Montalvo P.D., López-Batista V.F., Peluffo-Ordóñez D.H., Erazo-Chamorro V.C., Arciniega-Rocha R.P. (2019): “ Multivariate Approach to Alcohol Detection in Drivers by Sensors and Artificial Vision ”. In: Ferrández Vicente J., Álvarez-Sánchez J., de la Paz López F., Toledo Moreo J., Adeli H. (eds) From Bioinspired Systems and Biomedical Applications to Machine Learning. IWINAC 2019. Lecture Notes in Computer Science, vol 11487. Springer, Cham.

4.9.2. Objetivos

La Organización Mundial de la Salud, ha informado que el 40 % de todos los accidentes de tráfico son causados por el estado de ebriedad de los conductores. Además, es la quinta causa principal de muertes en las carreteras. Como resultado, 51 millones de personas resultan heridas o mueren cada año. Esto supone una pérdida por gastos de aproximadamente 500 millones de dólares a nivel mundial. En Ecuador, se registran 2100 accidentes de tránsito cada año por causas del alcohol. Lamentablemente, en los últimos 3 años este porcentaje ha aumentado, provocando un mayor número de pérdida de vidas y altos costes económicos para la sociedad. Esto se debe a que los efectos del alcohol en un conductor provocan alteraciones de la visión, la función psicomotora, cambios en la capacidad para reaccionar ante una alerta, comportamiento y conducta. En cuanto a las funciones psicomotoras, el tiempo de reacción del conductor aumenta. Esto afecta principalmente cuando el conductor necesita cambiar el pie del acelerador al freno, cuyo tiempo normal es 0,75 segundos, mientras que para un conductor en estado de ebriedad, el tiempo de reacción puede ser de 2 o más segundos. Como resultado, la probabilidad de sufrir un accidente de tráfico aumenta considerablemente.

El sistema propuesto se basa en la implementación y comparación de tres enfoques para la adquisición de datos del conductor. En este sentido, se propone utilizar un conjunto de sensores específicos: un sensor para medir la concentración de alcohol en el ambiente (tipo fisiológico), un sensor para capturar la temperatura de algunos puntos faciales del conductor (tipo biológico), y otro sensor capaz de identificar y reconocer el grosor de la pupila (tipo visual-característico). Ante esto, por un lado nuestro sistema busca eliminar la incertidumbre de la concentración de alcohol en la sangre. Por otro lado, el sistema se implementa en el interior del automóvil de forma no invasiva que permite reconocer al conductor y monitorear sus diferentes señales físicas y biológicas para determinar su idoneidad al volante. En consecuencia, es necesario realizar un acondicionamiento y calibración de la señal de cada sensor. Posteriormente, se implementa un análisis de datos permitiendo elegir el algoritmo adecuado, teniendo en cuenta la naturaleza de los datos. Finalmente, el sistema se evalúa con algunos criterios de rendimiento: (i) tasa de error, (ii) velocidad de clasificación del sistema y (iii) uso óptimo de los recursos del sistema integrado.

4.9.3. Metodología

El sistema está diseñado con las etapas de (i) establecimiento de requisitos del sistema, donde se calibran los sensores para su correcto funcionamiento y la toma de imágenes para la detección de la pupila. (ii) la adquisición de datos, para esto, se realizaron pruebas con varios conductores que presentaban diferentes estados de concentración de alcohol. Posteriormente, los conductores rinden una evaluación de detección de alcohol por parte de las entidades de control para validar la asignación de su grado de alcohol y (iii), la implementación de criterios de selección de prototipos unido a los algoritmos de clasificación más relevantes para la selección de los idóneos a ser implementados en el sistema.

4.9.4. Resultados

El algoritmo elegido para su implementación fue el *Support-Vector Machine* (SVM) en conjunto con el algoritmo *Condensed Nearest Neighbor* (CNN). Para validarlo, se realizaron 31 pruebas con varios conductores y diferentes estados de embriaguez. El sistema tuvo un rendimiento del 93,54 % con una sensibilidad del 100 %. Una especificidad del 85 % y una precisión del 89 %. Con la interacción de CNN en las siguientes 20 pruebas, se obtuvo un rendimiento del 95 %. Esto se debe a que la matriz de entrenamiento aumentó de valor con dos instancias más mejorando al sistema.

4.9.5. Conclusiones

El criterio implementado para la adquisición de datos del usuario fue correcto ya que permitió determinar satisfactoriamente su consumo de alcohol. Sin embargo, hubo algunas variaciones en la recopilación de datos debido a la variabilidad del entorno. Uno de ellos fue cuando el sistema estaba activo con la ventana del conductor abierta. Esto provocó que los gases circundantes, especialmente en zonas de mayor tránsito, el sensor reconociera una cierta cantidad de estos gases. La metodología propuesta para el respectivo análisis de datos fue adecuada para representar el evento en condiciones reales y el sistema puede tomar decisiones correctas en base a su experiencia. Por esta razón, CNN y SVM son los algoritmos óptimos para el conjunto de datos adquiridos. El sistema propuesto no resultó invasivo para el conductor y se puede implementar en otro tipo de vehículos sin mayores inconvenientes.

4.10. Contribución 6

4.10.1. Título

Rosero-Montalvo P.D., López-Batista V.F., Peluffo-Ordóñez D.H., Lorente-Leyva L.L., Blanco-Valencia X.P. (2019): “ Urban Pollution Environmental Monitoring System Using IoT Devices and Data Visualization: A Case Study ”. In: Pérez García H., Sánchez González L., Castejón Limas M., Quintián Pardo H., Corchado Rodríguez E. (eds) Hybrid Artificial Intelligent Systems. HAIS 2019. Lecture Notes in Computer Science, vol 11734. Springer, Cham.

4.10.2. Objetivos

Una de las principales causas del calentamiento global es la contaminación del aire por los efectos de la industrialización, la urbanización y el uso de vehículos de combustión. Esto se ha convertido en un gran riesgo para la salud en todos los países del mundo. La OMS estima que una de cada ocho muertes prematuras se debe a los efectos de las partículas contaminantes del aire. Además, cada año hay alrededor de 3 millones de personas mueren por contaminación del aire.

Para cumplir el objetivo de monitoreo de gases, la WSN fue instalada en diferentes sectores de la ciudad de Ibarra-Ecuador, donde se puede clasificar entre 3 casos: (i) altos niveles de contaminación, (ii) presencia moderada de gases y (iii) ausencia de emisiones. Para ello, 13 nodos de sensores fueron ubicados en la ciudad para enviar datos sobre las condiciones ambientales a través de la red de telefonía móvil 4G a un servidor IoT. El servidor permite la visualización de información sobre parámetros ambientales que muestran las horas y los lugares de mayor contaminación. Además, el usuario puede visualizar la situación de la ciudad mediante los colores del semáforo para una mayor comprensión. Debido a esto, en cada nodo WSN, se implementan criterios de selección de prototipos y clasificación supervisada para mejorar la tarea de decisión y evitar retrasos en la presentación de resultados.

4.10.3. Metodología

La metodología consiste con el desarrollo de la red WSN. Cada nodo sensor tiene un Arduino UNO con sensores MQ-7 (Monóxido de carbono), MQ-135 (Dióxido de carbono), ML8511 (Rayos UV), DTH11 (Temperatura y humedad). Además, de un panel solar, un gestor de batería y una batería tipo LiPo. Luego, se realiza una etapa de filtrado y suavizado de la señal de cada sensor. Posteriormente, se presenta un análisis de datos en dos etapas. La primera, una comparación con los algoritmos de selección de prototipos y balance de datos para encontrar nuevos conjuntos de datos de menor tamaño al volumen adquirido previamente. La segunda, con estos conjuntos de entrenamiento, se prueba con el algoritmo *K-Nearest Neighbors* (k-NN) para encontrar el mejor rendimiento del clasificador.

4.10.4. Resultados

El conjunto de entrenamiento obtenido por el algoritmo *Condensed Nearest Neighbor* (CNN) se ha elegido para mantener un alto rendimiento de clasificación. En cada nodo se ha implementado un algoritmo de k-NN y CNN. Por lo tanto, los nodos WSN envían solo una trama de datos depurada con la información de cada sensor y la etiqueta asignada a través de la red 4G hacia el servidor IoT. Como esquema final, se implementaron trece nodos que adquieren datos. En un principio, las primeras lecturas fueron realizadas sistemáticamente para lograr una ubicación óptima de los nodos WSN alrededor de la ciudad. Cada nodo se instaló al menos con 2 km de distancia de otro. Como resultado, fue posible mostrar gráficamente los sectores con mayor concentración de gases en tiempo real.

4.10.5. Conclusiones

El presente proyecto de monitoreo ambiental permitió conocer los lugares de contaminación dentro de la ciudad. Con esto, se pueden planificar diferentes estrategias para mitigar estos problemas. Con respecto al análisis de datos, la propuesta logró el objetivo de brindar la información correcta al clasificador y le permitió tomar una decisión adecuada en cada nodo WSN. Para ello, tecnologías como IoT permiten una adquisición ágil de datos e interconectan una gran cantidad de dispositivos para la extracción de conocimiento. Además, este análisis debe contar con una interfaz que le permita al usuario conocer las acciones realizadas por los nodos y poder realizar diferentes tipos de análisis con la información almacenada.

Con respecto a los análisis realizados, se pudo deducir que la concentración de gases nocivos puede afectar hasta un kilómetro a la redonda donde no existen emisiones. Además, los gases contaminantes pueden permanecer en altas concentraciones entre 25 y 30 minutos sin la existencia de vehículos que empeoren la situación. Finalmente, la neblina en lugares fríos permite una disipación de gases más rápida por la condensación del agua.

BIBLIOGRAFÍA

[1] M. C. Corporation, *Data Acquisition HandBook*, 2012.

[2] *Advanced Digital Signal Procesing and Noise Reduction*, 2008.

[3] P. Marwedel, *Embedded System Design*, 2018. [Online]. Available: http://link.springer.com/10.1007/978-3-319-56045-8

[4] W. J. Wang and C. H. Lin, "An Improved BitMask Based Code Compression Algorithm for Embedded Systems," in *2011 International Symposium on Electronic System Design*, dec 2011, pp. 152–157.

[5] N. Komatsu and M. Nakano, "Embedded Systems," in *Encyclopedia of Biometrics.* Boston, MA: Springer US, 2015, pp. 397–401. [Online]. Available: http://link.springer.com/10.1007/978-1-4899-7488-4_287

[6] Y. J. Park, J. Ahn, J. Lim, and S. H. Kim, ""C-chip" Platform for Electrical Biomolecular Sensors," in *Smart Sensors and Systems.* Cham: Springer International Publishing, 2015, pp. 3–23. [Online]. Available: http://link.springer.com/10.1007/978-3-319-14711-6_1

[7] G. Karsai, F. Massacci, L. J. Osterweil, and I. Schieferdecker, "Evolving embedded systems," *Computer*, vol. 43, no. 5, pp. 34–40, 2010.

[8] A. Canziani, E. Culurciello, and A. Paszke, "Evaluation of neural network architectures for embedded systems," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, may 2017, pp. 1–4.

[9] T. Bose, S. Bandyopadhyay, S. Kumar, A. Bhattacharyya, and A. Pal, "Signal Characteristics on Sensor Data Compression in IoT - An Investigation," in *2016 IEEE International Conference on Sensing, Communication and Networking (SECON Workshops)*, jun 2016, pp. 1–6.

[10] K. Kalantar-zadeh, "Introduction," in *Sensors.* Boston, MA: Springer US, 2013, pp. 1–9. [Online]. Available: http://link.springer.com/10.1007/978-1-4614-5052-8_1

[11] R. M. Ruairí, M. T. Keane, and G. Coleman, "A Wireless Sensor Network Application Requirements Taxonomy," in *2008 Second International Conference on Sensor Technologies and Applications (sensorcomm 2008).* IEEE, 2008, pp. 209–216. [Online]. Available: http://ieeexplore.ieee.org/document/4622664/

[12] A. M. Alajlan and K. M. Elleithy, "High-level abstractions in wireless sensor networks: Status, taxonomy, challenges, and future directions," in *Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education.* IEEE, apr 2014, pp. 1–7. [Online]. Available: http://ieeexplore.ieee.org/document/6820645/

[13] M. Jiménez, R. Palomera, and I. Couvertier, *Introduction to Embedded Systems.* New York, NY: Springer New York, 2014. [Online]. Available: http://link.springer.com/10.1007/978-1-4614-3143-5

[14] C. Meesookho, S. Narayanan, and C. S. Raghavendra, "Collaborative classification applications in sensor networks," in *Sensor Array and Multichannel Signal Processing Workshop Proceedings, 2002*, Aug 2002, pp. 370–374.

[15] H. Ayadi, A. Zouinkhi, B. Boussaid, and M. N. Abdelkrim, "A machine learning methods: Outlier detection in WSN," in *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, dec 2015, pp. 722–727.

[16] P. M. Reddy, "Embedded systems," *Resonance*, vol. 7, no. 12, pp. 20–30, dec 2002. [Online]. Available: http://link.springer.com/10.1007/BF02834526

[17] M. D. P. Emilio, "Data Acquisition Systems: Hardware," in *Data Acquisition Systems.* New York, NY: Springer New York, 2013, pp. 11–79. [Online]. Available: http://link.springer.com/10.1007/978-1-4614-4214-1_2

[18] C. Suarez, *Data acquisition handbook.* Clanrye Intl, 2015.

[19] B. Maag, Z. Zhou, and L. Thiele, "A Survey on Sensor Calibration in Air Pollution Monitoring Deployments," *IEEE Internet of Things Journal*, pp. 1–1, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8405565/

[20] C. Alippi, *Intelligence for Embedded Systems*, 2014. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84930809634&partnerID=tZOtx3y1

[21] W. Sung and J. Park, "Architecture exploration of a programmable neural network processor for embedded systems," *Proceedings - 2016 16th International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation, SAMOS 2016*, pp. 124–131, 2017.

[22] A. Ukil, S. Bandyopadhyay, A. Sinha, and A. Pal, "Adaptive Sensor Data Compression in IoT systems: Sensor data analytics based approach," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2015-Augus, 2015, pp. 5515–5519.

[23] L. Lin, X. Liao, H. Jin, and P. Li, "Computation offloading toward edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1584–1607, 2019.

[24] Y. W. Kim, S. J. Lee, G. H. Kim, and G. J. Jeon, "Wireless electronic nose network for real-time gas monitoring system," in *2009 IEEE International Workshop on Robotic and Sensors Environments*, 2009, pp. 169–172.

[25] S. Sadeghifard and L. Esmaeilani, "A new embedded e-nose system to identify smell of smoke," in *2012 7th International Conference on System of Systems Engineering (SoSE)*, 2012, pp. 253–257.

[26] J. Fu, J. Chen, Y. Shi, and Y. Li, "Design of a low-cost wireless surface emg acquisition system," in *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, 2013, pp. 699–702.

[27] I. E. Villalon-Turrubiates, "Classification algorithm for embedded systems using high-resolution multispectral data," in *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*, 2013, pp. 3582–3585.

[28] Y. Lei, W. Hongpeng, T. Dianxiong, and W. Jue, "A real-time hand gesture recognition algorithm for an embedded system," in *2014 IEEE International Conference on Mechatronics and Automation*, 2014, pp. 901–905.

[29] G. Surrel, F. Rincon, S. Murali, and D. Atienza, "Real-time probabilistic heart beat classification and correction for embedded systems," in *2015 Computing in Cardiology Conference (CinC)*, 2015, pp. 161–164.

[30] B. Sugiarto and R. Sustika, "Data classification for air quality on wireless sensor network monitoring system using decision tree algorithm," in *2016 2nd International Conference on Science and Technology-Computer (ICST)*, 2016, pp. 172–176.

[31] J. Lee, M. Stanley, A. Spanias, and C. Tepedelenlioglu, "Integrating machine learning in embedded sensor systems for internet-of-things applications," in *2016 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2016, pp. 290–294.

[32] F. Yang and L. Zhang, "Real-time human activity classification by accelerometer embedded wearable devices," in *2017 4th International Conference on Systems and Informatics (ICSAI)*, 2017, pp. 469–473.

[33] M. W. Tahir, N. A. Zaidi, R. Blank, P. P. Vinayaka, M. J. Vellekoop, and W. Lang, "An efficient and simple embedded system of fungus detection system," in *2017 International Multi-topic Conference (INMIC)*, 2017, pp. 1–4.

[34] S. Orguc, H. S. Khurana, K. M. Stankovic, H. S. Leel, and A. P. Chandrakasan, "Emg-based real time facial gesture recognition for stress monitoring," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2018, pp. 2651–2654.

[35] K. Belwafi, O. Romain, S. Gannouni, F. Ghaffari, R. Djemal, and B. Ouni, "An embedded implementation based on adaptive filter bank for brain–computer interface systems," *Journal of Neuroscience Methods*, vol. 305, pp. 1 – 16, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016502701830116X

[36] S. Aygun, E. O. Güneş, M. A. Subaşı, and S. Alkan, "Sensor fusion for iot-based intelligent agriculture system," in *2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, 2019, pp. 1–5.

[37] A. Marcu, G. Suciu, E. Olteanu, D. Miu, A. Drosu, and I. Marcu, "Iot system for forest monitoring," in *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, 2019, pp. 629–632.

[38] M. Levy and T. M. Conte, "Embedded multicore processors and systems," *IEEE Micro*, vol. 29, no. 3, pp. 7–9, May 2009.

[39] K. Kramer, T. Stolze, and T. Banse, "Benchmarks to find the optimal microcontroller-architecture," in *2009 WRI World Congress on Computer Science and Information Engineering*, vol. 2, March 2009, pp. 102–105.

[40] S. Edwards, L. Lavagno, E. A. Lee, and A. Sangiovanni-Vincentelli, "Design of embedded systems: formal models, validation, and synthesis," *Proceedings of the IEEE*, vol. 85, no. 3, pp. 366–390, March 1997.

[41] M. Levy and T. M. Conte, "Embedded multicore processors and systems," *IEEE Micro*, vol. 29, no. 3, pp. 7–9, May 2009.

[42] S. Mathew, "Advanced circuit techniques for high-performance microprocessor design challenges," in *IEEE International [Systems-on-Chip] SOC Conference, 2003. Proceedings.*, Sep. 2003, pp. 420–.

[43] P. D. Rosero-Montalvo, V. F. L. Batista, E. A. Rosero, E. D. Jaramillo, J. A. Caraguay, J. Pijal-Rojas, and D. H. Peluffo-Ordóñez, "Intelligence in embedded systems: Overview and applications," in *Proceedings of the Future Technologies Conference (FTC) 2018*, K. Arai, R. Bhatia, and S. Kapoor, Eds. Cham: Springer International Publishing, 2019, pp. 874–883.

[44] Y.-L. Lin, C.-M. Kyung, H. Yasuura, and Y. Liu, Eds., *Smart Sensors and Systems.* Cham: Springer International Publishing, 2015. [Online]. Available: http://link.springer.com/10.1007/978-3-319-14711-6

[45] H. Ghasemzadeh, S. Ostadabbas, E. Guenterberg, and A. Pantelopoulos, "Wireless Medical-Embedded Systems: A Review of Signal-Processing Techniques for Classification," *IEEE Sensors Journal*, vol. 13, no. 2, pp. 423–437, feb 2013.

[46] S. K. Kane, *Wearables.* London: Springer London, 2019, pp. 701–714. [Online]. Available: https://doi.org/10.1007/978-1-4471-7440-0_35

[47] P. D. Rosero-Montalvo, J. Pijal-Rojas, C. Vasquez-Ayala, E. Maya, C. Pupiales, L. Suarez, H. Benitez-Pereira, and D. H. Peluffo-Ordonez, "Wireless sensor networks for irrigation in crops using multivariate regression models," in *2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM)*, Oct 2018, pp. 1–6.

[48] A. M. Lopes, P. Abreu, and M. T. Restivo, "Analysis and pattern identification on smart sensors data," in *2017 4th Experiment@International Conference (exp.at'17).* IEEE, jun 2017, pp. 97–98. [Online]. Available: http://ieeexplore.ieee.org/document/7984409/

[49] K. Kalantar-zadeh, "Sensors ," in *Sensors.* Boston, MA: Springer US, 2013, pp. 11–28. [Online]. Available: http://link.springer.com/10.1007/978-1-4614-5052-8_2

[50] A. Nayyar and V. Puri, "A review of beaglebone smart board's-a linux/android powered low cost development platform based on arm technology," in *2015 9th International Conference on Future Generation Communication and Networking (FGCN)*, Nov 2015, pp. 55–63.

[51] R. Dasgupta and S. Dey, "A comprehensive sensor taxonomy and semantic knowledge representation: Energy meter use case," in *2013 Seventh International Conference on Sensing Technology (ICST)*, Dec 2013, pp. 791–799.

[52] K. Lian, C. Chiu, Y. Hong, and W. Sung, "Wearable armband for real time hand gesture recognition," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2017, pp. 2992–2995.

[53] Y. Gizlenmistir, "Filter based analysis unit design for data acquisition systems," in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, May 2018, pp. 1–4.

[54] P. Kowalski and R. Smyk, "Review and comparison of smoothing algorithms for one-dimensional data noise reduction," in *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, May 2018, pp. 277–281.

[55] B. Araujo, *Aprendizaje automático : conceptos básicos y avanzados : aspectos prácticos utilizando el software Weka*, ser. Pearson Educación. Pearson Prentice Hall, 2006. [Online]. Available: https://books.google.com.ec/books?id=BCzUAQAACAAJ

[56] P. C. Sen, M. Hajra, and M. Ghosh, “Supervised classification algorithms in machine learning: A survey and review,” in *Emerging Technology in Modelling and Graphics*, J. K. Mandal and D. Bhattacharya, Eds. Singapore: Springer Singapore, 2020, pp. 99–111.

[57] P. D. Rosero-Montalvo, V. F. López-Batista, D. H. Peluffo-Ordóñez, V. C. Erazo-Chamorro, and R. P. Arciniega-Rocha, “Multivariate approach to alcohol detection in drivers by sensors and artificial vision,” in *From Bioinspired Systems and Biomedical Applications to Machine Learning*, J. M. Ferrández Vicente, J. R. Álvarez-Sánchez, F. de la Paz López, J. Toledo Moreo, and H. Adeli, Eds. Cham: Springer International Publishing, 2019, pp. 234–243.

[58] U. Martinez-Corral and K. Basterretxea, “A fully configurable and scalable neural coprocessor IP for SoC implementations of machine learning applications,” in *2017 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, jul 2017, pp. 125–132.

[59] P. D. Rosero-Montalvo, D. F. Peña-Unigarro, D. H. Peluffo, J. A. Castro-Silva, A. Umaquinga, and E. A. Rosero-Rosero, “Data visualization using interactive dimensionality reduction and improved color-based interaction model,” in *Biomedical Applications Based on Natural and Artificial Computing*, J. M. Ferrández Vicente, J. R. Álvarez-Sánchez, F. de la Paz López, J. Toledo Moreo, and H. Adeli, Eds. Cham: Springer International Publishing, 2017, pp. 289–298.

[60] M. Verleysen and J. A. Lee, “Nonlinear dimensionality reduction for visualization,” in *Neural Information Processing*, M. Lee, A. Hirose, Z.-G. Hou, and R. M. Kil, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 617–622.

[61] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim, “Visual interaction with dimensionality reduction: A structured literature analysis,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 241–250, Jan 2017.

[62] L. J. P. Van Der Maaten, E. O. Postma, and H. J. Van Den Herik, “Dimensionality Reduction: A Comparative Review,” *Journal of Machine Learning Research*, 2009.

[63] L. Sun and M.-V. MISSING-VALUE, “Nonlinear Dimensionality Reduction,” in *Multi-Label Dimensionality Reduction*, 2018.

[64] Y. Zhai, W. Song, X. Liu, L. Liu, and X. Zhao, “A chi-square statistics based feature selection method in text classification,” in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, Nov 2018, pp. 160–163.

[65] M. Dash and H. Liu, “Feature selection for classification,” *Intelligent Data Analysis*, vol. 1, no. 1, pp. 131 – 156, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1088467X97000085

[66] M. Peker, A. Arslan, B. Sen, F. V. Celebi, and A. But, “A novel hybrid method for determining the depth of anesthesia level: Combining relieff feature selection and random forest algorithm (relieff+rf),” in *2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA)*, Sep 2015, pp. 1–8.

[67] A. Mathur and G. M. Foody, “Multiclass and binary svm classification: Implications for training and classification users,” *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 2, pp. 241–245, April 2008.

[68] K. Kang, F. Gao, and J. Feng, “A new multi-layer classification method based on logistic regression,” in *2018 13th International Conference on Computer Science Education (ICCSE)*, Aug 2018, pp. 1–4.

[69] M. S. Majdi, S. Ram, J. T. Gill, and J. J. Rodríguez, “Drive-net: Convolutional network for driver distraction detection,” in *2018 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*, April 2018, pp. 1–4.

[70] M. Hossin and M. Sulaiman, “A review on evaluation metrics for data classification evaluations,” *International Journal of Data Minning & Knowledge Managment Process*, 2015. [Online]. Available: http://aircconline.com/ijdkp/V5N2/5215ijdkp01.pdf

Parte V

ANEXO DE CONTRIBUCIONES

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2018.2885323, IEEE Sensors Journal

Intelligent System for Identification of Wheelchair User's Posture using Machine Learning Techniques

Paul Rosero-Montalvo[1,2,6], Diego H. Peluffo-Ordóñez[3,4,5], Vivian F. López Batista[2], Jorge Serrano[3] and Edwin Rosero[1,6]
[1] Universidad Técnica del Norte - Ecuador,
[2] Universidad de Salamanca-Espanã,
[3] School of Mathematical Sciences and Information Technology, Yachay Tech - Ecuador,
[4] Corporación Autónoma de Nariño, Pasto - Colombia,
[5] Universidad de Nariño - Colombia,
[6] Instituto Tecnológico Superior 17 de Julio- Ecuador.

***Abstract*—This work presents an intelligent system aimed at detecting a person's posture when sit a wheelchair. The main use of the proposed system is to warn an improper posture to preventing major health issues. A network of sensors is used to collect data that are analyzed through a scheme involving the following stages: Selection of prototypes using Condensed Nearest Neighborhood rule (CNN), data balancing with the Kennard-Stone (KS) algorithm, and reduction of dimensionality through Principal Component Analysis (PCA). In doing so, acquired data can be both stored and processed into a micro controller. Finally, to carry out the posture classification over balanced, pre-processed data, the K-Nearest Neighbors (k-NN) algorithm is used. It turns to be an intelligent system reaching a good trade-off between the necessary amount of data and performance is accomplished. As a remarkable result, the amount of required data for training is significantly reduced while an admissible classification performance is achieved being a suitable tradegiven the device conditions.**

***Index Terms*—embedded system, kennard-stone, K-Nearest Neighbors, principal component analysis, posture detection.**

I. Introduction

People using wheelchairs may have either a physical, mental and/or sensory disability that limits their everyday activities. Around the world, approximately 15% of the population has some type of mobility problem [1] [2] and market research studies forecast a growth in manual wheelchair expenditure from $ 1.8 billion US in 2011 up to $ 2.9 billion US in 2018 [3]. While using wheelchair has been shown to increase the quality of life of users by enabling mobility, occupation and social interaction, among others[4], [5], [6], the health condition of wheelchair users is strongly affected by sitting posture. In particular, bad posture on a wheelchair leads chronic pain, sclerosis, kyphosis, skin and respiratory problems, loss of brain skills, some physical health problems such muscle rigidity, fatigue and muscle pain, among others [7][8]. The causes can be: lack and imbalance muscle, lack of exercise, bad position of wheelchair pillows and usual functional activities carried out in the same way every day. In some cases the user lacks a feeling of pain until its health has been seriously compromised [9]. On the other hand, multiple benefits are achieved when the user shows the habit of sitting right: pain intensity decreases and the probability of ulcers formation is reduced [10]. Among the different parts of the wheelchair, the seat stands out in providing health benefits. It determines the users stability and distributes uniformly the users weight on the largest possible area [7]. When the seat is shorter than what the user needs, an increase in pressure is applied on the buttocks. On the contrary, when it is longer that needed, an increment of pressure is then applied in the knees. Unbalanced seat bending will make the user feel a lack of symmetry and cause the thighs and knees to be pushed. This in turn will result in excess of pressure and friction, affecting in long term the hip, as displayed in Fig. 1.

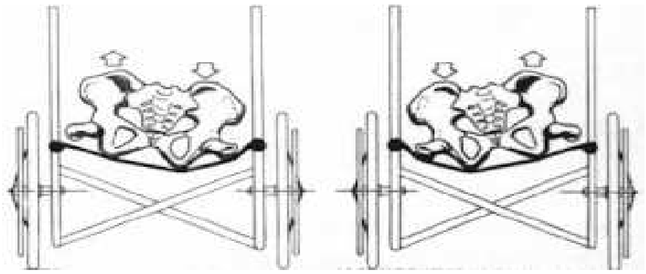


Fig. 1. Asymmetrical stress on hips due to unbalanced sitting posture on a wheelchair.

When the user holds an adequate sitting posture, the hip angle formed by the thighs and the trunk stabilizes the pelvis. An angle of 90° has been found optimal for most of everyday activities. The best way to achieve this angle is by using a customized cushion, adapted to the human shape, located behind low back in order to accommodate the buttocks shape.

Some related works [10], and [7] have developed studies on the manners that people typically sit and their consequences. But they lack intelligent posture classification solutions. A related study, [11] performs a similar work of posture selection, but over conventional chairs and therefore does not take into account the affections of hip imbalance. In the context of sensor use and configuration, [12] and [13] explain main uses and applications of a broad variety of sensors. Albeit not being researches focused on the detection of postures, they do provide with guidelines for the selection, location and reading ways of sensors. Also, other studies [14], [15] have proved the benefit of using k-NN and sensors to classify human activities, mainly in the detection of sign language, reaching a classification performance around 80% (under ideal non-realistic environments). In addition, they stress the fact that using k-NN in electronic systems results adequate given the computational limitations of the processor.

In this paper, we present an intelligent system to inform in real time the wheelchair user about the correct sitting posture. The system is based on a pressure sensor network embedded in the seating and back cushion of the wheelchair, used to acquire the position related variables, and on classification machine learning techniques. To design our system, we took with various commercial wheelchair types and we integrated the sensors in the seat and the backrest for data acquisition. This allows for determining the user sitting posture with high accuracy [16]. The position readings were collected in a high-dimensionality database, having reading variations/errors on raw data. The data were filtered by performing a prototype selection procedure through a Condensed Nearest-Neighbours (CNN) algorithm [17]. Subsequently, the Kennard-Stone (KS) method was employed to balance the classes' distribution into the database, as well as a Principal Component Analysis (PCA) process was applied to further reduce the number of variables to feed the microcontroller [18]. Finally, a k-NN automatic learning algorithm was implemented to classify position readings. On doing so the final user will be warned in case deviations from the optimal, on-wheelchair sitting posture. The k-NN classification algorithm activates through the micro controller a series of vibrators that signal to the wheelchair user the areas in the seating displaying pressure excess, thus allowing for a posture correction and enhancing the userś health state. The system has -in average- a high level of accuracy in classification performance.

The remaining of the paper is organized as follows: Section 2 presents the electronic design, database settings and data analysis. Section 3 holds the conducted tests and obtained results. Finally, Section 4 gathers the final remarks as conclusions and future work.

II. Materials and methods

The intelligent embedded system is designed to be conformed by three main stages: (a) design of the electronic device for location and data acquisition using sensors, (b) database configuration, and (c) data analysis, namely, prototype selection, classes balancing, dimensionality reduction, and classification algorithm.

A. Design of the electronic device

Fig. 2 shows a wheelchair view displaying the pressure sensors (red circles) and the vibrator devices (blue circles) on the seat, as well as the micro-controller and another pair of sensors (grey circles) at the backrest. The figure also shows the hardware connection and the incoming and outgoing communication between the microprocessor and sensors/actuators [19], displayed by red and yellow arrows, respectively.

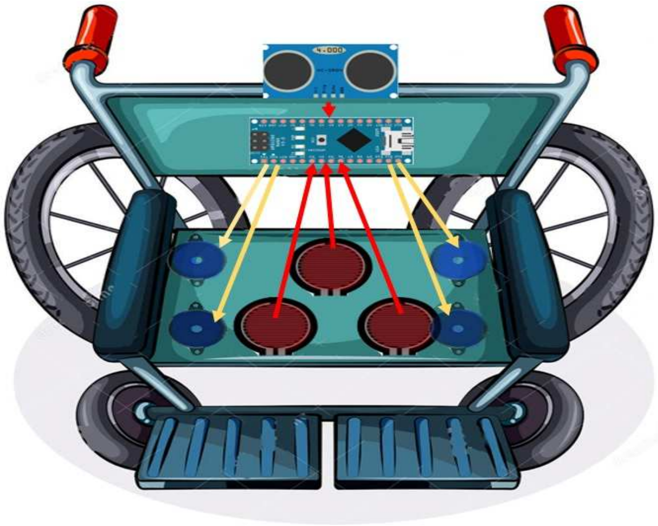


Fig. 2. A view of the wheelchair displaying the embedded system using diagram blocks for the sensors and actuators. Pressure position sensors and actuators are represented by red and blue circles on the seating, respectively. Black circles display ultrasound sensors in the backrest, yellow arrows show output peripheral, and red arrows indicate peripheral input into the analog-digital converter.

Three pressure sensors are employed for data acquisition and located inside the seat filling. Their position is selected to fit the ideal place just under the coccyx and the legs. Besides, an ultrasonic sensor was used in order to determine the distance between the back and the wheelchair backrest. The analog signals obtained from the sensors were digital converted with an Arduino nano micro-controller with Ohm's law of resistance sensing technique [20]. Table I shows the description of each electronic element.

In this work, sensor location was carried out by performing an electronic, data reading analysis, which consists of: (i) Voltage loss analysis in each sensor, and (ii) data readout stabilization by analog conversion between 0 to 1023.

As final important considerations on how to implement the electronic system, we highligth the following ones: (i) It is implemented in a single system within a wheelchair, communication between sensors, actuators and the system is done with connection cables. A LiPo battery of 4.7 volts and 700 mA is initially used. (ii) A library of k-NN classification algorithms was developed, which is called after having received 10 readings from the sensors and finding their average to avoid errors. (iii) Due to the system conditions, the library runs parallel to the RAM of the system. Therefore in some cases it is necessary to perform general resets to avoid saturating the system. (iv) The system uses approximately 70% of the available memory of the micro controller (32KB available), this percentage varies in relation to the training matrix stored in the system.

TABLE I
ELECTRONIC ELEMENTS DESCRIPTION

Element	Description	Available at
Pressure sensor	This is a force sensor with a round, 0.5” diameter, sensing area. This sensor will vary its resistance depending on how much pressure is being applied to the sensing area. The harder the force, the lower the resistance.	https://www.sparkfun.com /products/9375
Arduino nano	The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.x)	https://store.arduino.cc/ usa/arduino-nano
Ultrasonic Sensor	This is the HC-SR04 ultrasonic ranging sensor. This economical sensor provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm	https://www.sparkfun.com /products/13959
Vibration Motor	With a 2-3.6V operating range, these units shake crazily at 3V	https://www.sparkfun.com /products/8449

B. Database settings

To ascertain the typical sitting postures on a wheelchair in order to label the acquired readings (henceforth called samples) for solving our classification problem, we take advantage of a conventional taxonomy recommended by physicians and physiotherapists expert at this topic. As a result, four common sitting postures together with the effect on the backbone were identified, as depicted in Fig. 3 and Table II describes their related potential health issues.

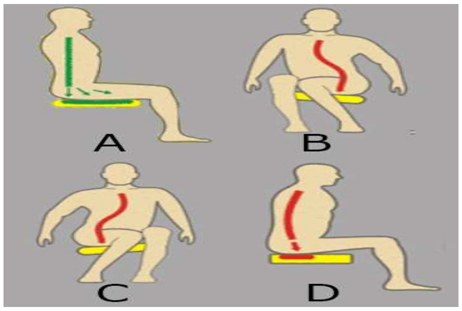


Fig. 3. Most common on-wheelchair postures: their effects at the spinal column and weight distribution on the seat surface.

The data acquisition protocol utilized in this work is as follows: We consider 5 individuals asked to keep every single selected posture (A, B, C and D) during two minutes (average time needed to reach a reliable sample). When the system detects a bad posture in a wheelchair, it waits for a 2 minutes. In the absence of new information, the system activates the actuators inside the seat that indicate through vibration the points where there is an excess of pressure. Upon activation of this alarm signal, the user is pointed to change sitting posture until correcting the pressure distribution exerted on the seat. At that moment the vibrators power supply is switched off.

TABLE II
LABELS AND EFFECTS OF DIFFERENT SITTING POSTURES ON A WHEELCHAIR

Position label	Description	Possible health problems
A	Right position	No harm
B	Higher pressure on right side	Respiratory issues, muscle imbalance stress on liver, stomach and right kidney
C	Higher pressure on left side	Respiratory issues, muscle imbalance stress on spleen and left kidney
D	Higher forward pressure	Knee issues, back pain, and stress on abdomen

Each individual repeated 25 trials per posture. Then, since 4 sensors are used, a total of 2000 data points (500 samples × 4 sensors) are obtained.

C. Data analysis

The acquired data are stored into a matrix $\boldsymbol{Y} \in \mathbb{R}^{m \times n}$ order, where **m** is the number of samples and $\boldsymbol{n}$ amounts the number of sensors (that is to say, the number of attributes representing each sample). Meanwhile, $\boldsymbol{L} \in \mathbb{R}^{m \times 1}$ is a vector holding the sample labeling. In this case, $\boldsymbol{m}$=500 and $\boldsymbol{n}$ = 4. Proposed data analysis framework involves the following stages: (1) prototype selection using CNN, (2) data balancing with KS algorithm, (3) dimensionality reduction via PCA, and (4) data classification. An explaining block diagram of the data analysis process is shown in Fig. 4.

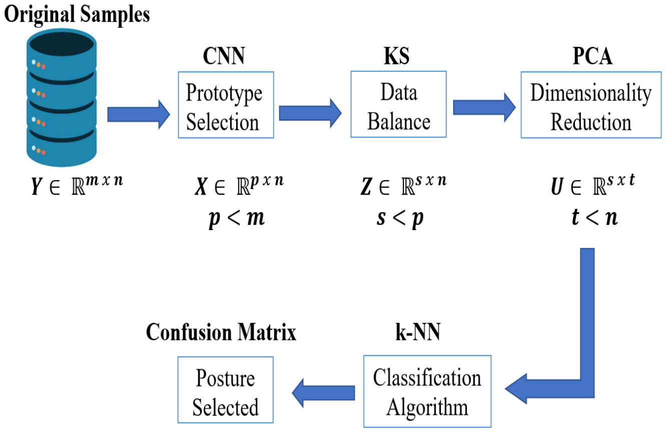


Fig. 4. Data analysis framework: it starts with the raw database matrix $\boldsymbol{Y}$. Then, such a matrix undergoes a progressive dimension reduction by using prototype selection and data balance (reducing the number of samples) as well a data representation procedure (reducing the number of attributes). Finally, a classification algorithm is used to assign a final posture label.

Following are explained the considered stages:

1) Prototype selection: As well-known, in pattern recognition, supervised classifiers need a labeled dataset of information to feed the algorithms in the training stage. Due to limitation in computational resources, resource optimization is a major issue to ensure proper functioning in embedded systems [21]. In practice, not all data are useful, therefore irrelevant data should be discarded. This process is called *prototype selection*, and allows for reducing the size of the training

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2018.2885323, IEEE Sensors Journal

data set. A remarkable advantage of prototype selection is the decrease of execution times, space complexity, and computational cost, besides eliminating noise [22]. Works as those presented in [23][24] use prototype selection in many data sets reaching training set size reduction about 87% and 97% of the total amount of instances, respectively. Besides, they explain how the combination PS with classifiers algorithms works in a multi-class environment. Along with this, noise elimination algorithms are studied, which a are aimed at filtering boundary points staying out of pre-established neighborhoods, and thus yielding soft decision boundaries [23].

Our choice for prototype selection algorithm follows from the so-called CNN. It algorithm determines two subgroups, called $\boldsymbol{S}$ (training set) and $\boldsymbol{T}$ (test set), and it eliminates the data that the algorithm cannot properly classify [22]. Indeed, similar k-NN based approaches have been used successfully in other applications such as gases analysis [25]. In practice, prototype selection allow us to reduce the noise between data assigned to the four labels, A to D, corresponding to the 4 different sitting postures displayed in Fig. 2 and table I, and obtain the matrix $\boldsymbol{X} \in \mathbb{R}^{p\times n}$ with $\boldsymbol{p}$ less than $\boldsymbol{m}$.

2) Data balance regarding classes: An important issue faced at data acquisition stages when using embedded system is the ability to balance uniformly the training data for each label. Data acquisition rate is strongly correlated to the sensor reading speed, highly dependable on the wheelchair user action. KS algorithm allows balancing the data set in relation to the four labels [26]. We thus obtain the lower dimensionality matrix $\boldsymbol{Z} \in \mathbb{R}^{s\times n}$. Despite this data reduction, there are still too much data to be fed into the micro controller ($\boldsymbol{s}$ is less than $\boldsymbol{p}$).

3) Dimensionality reduction: Representing a set of high dimensional data increases the complexity in the user's understanding and the information can become abstract and intricate [18]. Dimensionality reduction (DR) is one of the approaches to convert the data into a simpler and more compact way. DR methods thus enable to represent large volumes of information at optimal processing times, maintaining the same properties of complex high-dimensional data. One of these methods is the so-called PCA, that makes a projection on the variables that can better represent the data set in terms of least squares fits[27]. Application of PCA algorithm to $\boldsymbol{Z}$ matrix further reduces the volume of information and results into a lower order matrix called $\boldsymbol{U} \in \mathbb{R}^{s\times t}$ with $\boldsymbol{t}$ less than $\boldsymbol{n}$.

4) Classification: Finally, a classification algorithm is implemented. k-NN is one of the non-parametric algorithms most used for machine learning due to its simplicity and effectiveness, considered one of the 10 best methods of automatic learning [23]. k-NN is simple to implement since it uses the euclidean distance between the entire training set to sort new incoming data. Its aim is to isolate a small group of information, which allows predicting its class with the same quality as the initial set of data. k-NN algorithm has been shown to be computationally effective for this system when analyzing an non-hardly-separable-classes database (as that obtained with the aforementioned dimension reduction stages) [19]. The new incoming data classification is then performed by a k-NN algorithm in real time when wheelchair users are requested to sit down adopting one of the four different postures.

III. Results

To assess the behavior and benefit of every single system stage, we will first discuss over some results related to the dimension reduction (regarding the used training set) procedure. Then, the outcomes of the classification algorithm (in terms of both performance and processing time) are described to state the whole system performance.

A. Training set reducing results

The proposed system yields a considerable reduction of the training matrix (namely, 88%) while keeping the same classification performance. In this way, the micro-controller can perform the classification process in optimal times (less than one minute) without overburdening its resources such as RAM, work registers, cycle counters, among others. It should be mentioned that the classifier has some drawbacks from the data acquisition form. To overcome such drawbacks, the classification process is performed once the sensors are installed in the wheelchair, and users are asked for adopting postures being not ideal to acquisition data. As a result, the training matrix becomes highly noise. The dimensions of input database matrix $\boldsymbol{Y}$ are $\boldsymbol{m} = 500$ and $\boldsymbol{n} = 4$. Upon application of prototype selection using the CNN algorithm with neighbor $\boldsymbol{k} = 1$, the resultant matrix $\boldsymbol{X}$ is $\boldsymbol{p} \times \boldsymbol{n}$ dimensional, with $\boldsymbol{p} = 260$. Subsequently, the balance data stage using the KS algorithm results in a matrix $\boldsymbol{Z}$ in size of $\boldsymbol{s} \times \boldsymbol{n}$, being $\boldsymbol{s} = 80$.

Fig. 5 shows the scatter plot for the original data matrix at left, and the matrix $\boldsymbol{Z}$ at right. Marker colors are given correspondingly to the established four labels (the four considered sitting postures).

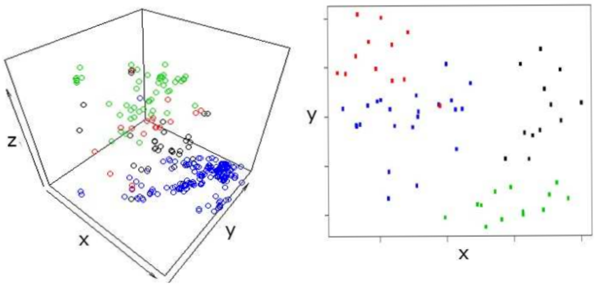


Fig. 5. Scatter plots for original data matrix $\mathbf{Y}$ (at left) and reduced data matrix $\mathbf{U}$ (at right). Scattering colors correspond to the four characteristic, sitting postures. Colors green, blue, red, and black correspond respectively to posture label A, B, C, and D.

Table III holds the PCA summary in terms of the component indicators, namely, the standard deviation, proportion of variance and cumulative proportion. A significantly smaller value for the standard deviation is observed for PC4, which corresponds to a sitting posture with the back leaning forward in the wheelchair. We attribute this reduction in the deviation to the lack of data stemming from the ultrasound sensors at the backrest of the wheelchair. As will be further discussed , the lack of these data will impact the accuracy of the prediction performance by the classification algorithm for this sitting posture. As a final PCA result, we obtain a matrix $\boldsymbol{U} \in \mathbb{R}^{s \times t}$, with $\boldsymbol{t} = 2$.

TABLE III
PCA SUMMARY METHOD APPLIED TO $\mathbf{Z}$ MATRIX

Component indicators	PC1	PC2	PC3	PC4
Standard deviation	223.66	171.77	112.89	11.24
Proportion of variance	0.5414	0.3193	0.1379	0.0014
Cumulative proportion	0.5414	0.8607	0.9986	1.00

Following from the the standard deviation weight given by the applied PCA procedure, the equations to reduce of 4 dimension to a 2 dimension ($\boldsymbol{x}$ and $\boldsymbol{y}$ axes) representation are:

$$\mathbf{x} = \sum_{i=0}^{\mathbf{s}}(0.6125 * i) + (0.5228 * i) - (0.592 * i) + (0.019 * i), \tag{1}$$

$$\mathbf{y} = \sum_{i=0}^{\mathbf{s}}(0.7724 * i) - (0.554 * i) - (0.309 * i) + (0.008 * i), \tag{2}$$

being i the index for rows.

B. Classification algorithm performance

We tested the k-NN algorithm with all training matrix generated within the proposed methodology ($\boldsymbol{Y}$, $\boldsymbol{X}$, $\boldsymbol{Z}$, and $\boldsymbol{U}$). The corresponding classification accuracy reached in a cross validation (10 iterations) fashion are shown in Table IV .

TABLE IV
CLASSIFICATION PERFORMANCE USING DIFFERENT TRAINING SET

Training Set representation	Pos. A (mean ± std)	Pos. B (mean ± std)	Pos. C (mean ± std)	Pos. D (mean ± std)
matrix $\mathbf{Y}_{(500\times4)}$	85.45 ± 4.25	82.28 ± 4.8	78 ± 5.68	63.5 ± 8.2
matrix $\mathbf{X}_{(260\times4)}$	88.45 ± 3.28	80.21 ± 4.12	78 ± 4.32	65.5 ± 5.11
matrix $\mathbf{Z}_{(80\times4)}$	83.28 ± 2.45	80.21 ± 3.75	78 ± 3.77	66.5 ± 4.89
matrix $\mathbf{U}_{(80\times2)}$	81.58 ± 1.89	77.50 ± 2.14	78 ± 2.55	67.14 ± 3.04

It is worth noticing the significance reduction reached in the classification (also, prediction) power of label D with respect to the other labels. This is tentatively assigned to the lack of information of the ultrasound sensors corresponding to such a posture. After applying the proposed methodology (the electronic device, and data analysis), an overall average performance of 76.05% is observed. Upon setup on the wheelchair and real time classification, a value of 75.2% prediction accuracy in posture detection was obtained for the the embedded system as is summarized in Table V.

As notable remarks about the Table V, it is important to mention the subsequent ones: The embedded system cannot storage the matrix **Y**, since its size is bigger than flash memory available capacity. The matrix $\boldsymbol{X}$ consumes a lot of memory resources causing that the system some times does not response. With matrix **Z**, the system works at a comparable performance than the one reached with the complete database. Even better with matrix $\boldsymbol{U}$ as the battery consumption is significantly lower. The battery consumption was made at the detection of 10 postures.

TABLE V
USED MEMORY AND CONSUMED ELECTRICAL CURRENT TESTS

Training Set representation	Used % Memory	Exec. time Simulation	Exec. Time Real	Current
matrix $\mathbf{Y}_{(500\times5)}$	NN	NN	NN	NN
matrix $\mathbf{X}_{(260\times5)}$	95	1.71s	3.5s	359mA
matrix $\mathbf{Z}_{(80\times5)}$	65	1.25s	2.2s	267mA
matrix $\mathbf{U}_{(80\times2)}$	42	1.1s	1.5s	202mA

By reducing the memory consumption, the program-counter-records reading cycles become significantly limited, while the embedded systems' lifetime is increased. Another advantage, given the computational-cost lowering, system can be switched to sleep mode more properly, as their records are not focused any longer on reading a large amount of data and may couple better with these programming platforms. Finally, Table V shows the execution times of the simulation and the real implementation of the system. It must be considered that the value of the simulated time can be affected by the characteristics of the higher-performance computer where such a program is running. When working in real conditions, the execution time increases considerably. This is because the system to function better has an internal reset that deletes (re-initializes) the program counters -which keep so until new data is provided by the sensors. This process, on the one hand, helps to get a better reading and comparison among new information. And on the other hand, it generates a response delay of the system due to the process itself execution. With the matrix X, the reset process is affected considerably and causes the system to have a longer reaction time to the smaller matrices. In this sense, it must be considered that regarding the user perception (vision) the system does not have a significant response time between one data set and another. Nonetheless, at a device level, accordingly to the internal clock of 16 MHz, it can mean that there are thousands of lines of work in such a small-time interval.

An essential goal for this work is reducing as much as possible the magnitude necessary of resources for carrying out the embedded system tasks. Then, in order to set a reference point, a comparison -in terms of computational performance-with a higher performance system (in this case, a standard PC) is performed. To that purpose, we use a mathematical formula able to quantify the functionality embedded system behavior. Let f be a real or complex valued function and g a real valued function, both defined on some unbounded subset of the real positive numbers, such that $g(x)$ is strictly positive for all large

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2018.2885323, IEEE Sensors Journal

enough values of x, then it is satisfied that:

$$f(x) = O(g(x)), \tag{3}$$

$$O(g(x)) = \sum_{m=1}^{M}\sum_{k=1}^{K}\frac{1}{\alpha_m}\log\frac{x}{k\beta_k}, \tag{4}$$

where $O(\cdot)$ is a complexity operator, $x \longrightarrow \infty$, $\{\alpha_1, ..., \alpha_M\}$ is the set of weighting factors. To make selection of weighting factors intuitive, we use probability values so that $0 <= \alpha_m <= 1$. Likewise, $\{\beta_1, \ldots, \beta_K\}$ are performance variables, therefore they are used for normalization purposes, so that the importance is presented in a decreasing order regarding their values. A detailed explanation of this measure is presented in [28].

In terms of the measure $f(x)$, our system is compared against a standard computer (processor core I7, RAM 10Gb, 1 Tb of memory). The tests are carried out with matrices $\boldsymbol{X}$, $\boldsymbol{Z}$ and $\boldsymbol{U}$. Matrix $\boldsymbol{Y}$ is discarded since it cannot be directly used to feed our system. Nonetheless, by taking advantage of its computational capabilities, over the standard PC the classification procedure is performed with the matrix $\boldsymbol{Y}$. The Fig. 6 shows the resulting plotting of $f(x)$ for each considered matrix. The axis x means time execution of the system and y axis means the performance in scale 1 to 10.

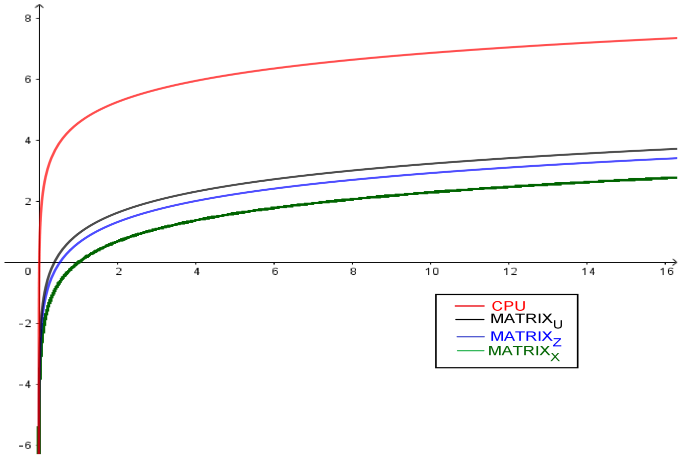


Fig. 6. Funcionality performance comparison between a standard PC (CPU) and the proposed embedded system tested over different representation matrices.

As naturally expected, the standard PC outperforms the proposed system outcomes when testing the reduced matrices, however, the standard PC battery is very consumed for this algorithm and others task the system makes. Notwithstanding, it can clearly observed how the systems when run over matrix $\boldsymbol{U}$ not only resembles the shape of the functionality curve of the reference PC but overcomes the rest of system runnings (using either $\boldsymbol{Z}$ or $\boldsymbol{X}$). This fact further justifies the benefit of the use of techniques for reducing the needed training set size.

IV. Conclusions

In this work, we have proposed the design of an embedded system incorporating a k-NN classification algorithm inside the RAM memory, which, along with proper training-set resizing stages, is able to detect sitting posture defects on a wheelchair reaching an average accuracy of over 75%. Additionally, our system achieves a significant reduction in dimensionality, as well as reduces the amount of required training-data by 88%. Particularly, this is reached by applying CNN as a prototype selection algorithm, a KS data balance algorithm, and PCA. Thus, this minimal data representation enables our system to both storage and classify some sitting postures in real time by means of any simple classification algorithm (for instance, k-NN algorithm as done in this study).

The performance of the system was considered acceptable given its computational resources and context conditions. An important characteristic to be highlighted is that the system responds to poor posture in approximately 1 second, which enables the generation of an adequate warning to the user when seated in a bad posture.

Our experimental results demonstrate that the wheelchair users' quality of life can be notably enhanced through implementing cost-effective embedded-systems-based solutions powered by machine learning.

A remarkable aspect to be considered for further research is the use of CSV format files for readings within the electronic system. Such a format is easy to read and demands low resource consumption being suitable for intelligent systems aimed at making decisions rapidly (i.e. real time applications). As well, the incorporation of an extra sensor to improve the system performance is considered, especially regarding location D to minimize errors at identifying postures B and C.

Acknowledgments

We are in debt to J. C. Moreno for a critical reading of the manuscript. The authors thank the Universidad Técnica del Norte and the Instituto Tecnológico 17 de Julio-Yachay for kindly providing access to their electronics laboratories to perform the system tests. As well, we especially thank to SDAS Research Group (www.sdas-group.com) for its valuable support.

References

[1] Y. R. Huang and X. F. Ouyang, "Sitting posture detection and recognition using force sensor," in *2012 5th International Conference on BioMedical Engineering and Informatics*, Oct 2012, pp. 1117–1121.

[2] H. Ishimatsu and R. Ueoka, "Bitaika: Development of self posture adjustment system," in *Proceedings of the 5th Augmented Human International Conference*. New York, NY, USA: ACM, 2014, pp. 30:1–30:2.

[3] W. Research, *Manual Wheelchair Market Shares, Strategies, and Forecasts, Worldwide, 2012 to 2018.* WinterGreen Research, 2012, vol. REPORT # SH24912312.

[4] M. Devitt, B. Chau, and J. Jutai, "The effect of wheelchair use on the quality of life of persons with multiple sclerosis," *Occupat. Ther. Health Care*, vol. 17, pp. 63–79, 2004.

[5] I. Pettersson, K. Törnqueist, and G. Ahlström, "The effect on an outdoor powered wheelchair on activity and participation in users with stroke," *Disabil. Rehabil.: Assist Technol.*, vol. 1, pp. 235–243, 2006.

[6] W. B. Mortenson, W. C. Miller, J. Boily, B. Steele, L. Odell, E. M. Crawford, and G. Desharnais, "Perceptions of power mobility use and safety within residential facilities," *Can. J. Occupat. Ther.*, vol. 73, pp. 142–152, 2005.

[7] G. Liang, J. Cao, X. Liu, and X. Han, "Cushionware: A practical sitting posture-based interaction system," in *CHI '14 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2014, pp. 591–594. [Online]. Available: http://doi.acm.org/10.1145/2559206.2574778

[8] B. Mutlu, A. Krause, J. Forlizzi, C. Guestrin, and J. Hodgins, “Robust, low-cost, non-intrusive sensing and recognition of seated postures,” in *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST ’07. New York, NY, USA: ACM, 2007, pp. 149–158.

[9] Y. H. Wu, C. C. Wang, T. S. Chen, and C. Y. Li, “An intelligent system for wheelchair users using data mining and sensor networking technologies,” in *2011 IEEE Asia-Pacific Services Computing Conference*, Dec 2011, pp. 337–344.

[10] Y. Mingjiu, Y. Jun, Z. Quan, and L. Changde, “Ergonomics analysis for sitting posture and chair,” in *2006 7th International Conference on Computer-Aided Industrial Design and Conceptual Design*, Nov 2006, pp. 1–4.

[11] P. Rosero-montalvo, D. Jaramillo, S. Flores, D. Peluffo, V. Alvear, and M. Lopez, “Human Sit Down Position Detection Using Data Classification and Dimensionality Reduction,” *Advances in Science, Technology and Engineering Systems Journal*, vol. 2, no. 3, pp. 749–754, 2017. [Online]. Available: http://astesj.com/v02/i03/p95/

[12] A. Nag, S. C. Mukhopadhyay, and J. Kosel, “Wearable flexible sensors: A review,” *IEEE Sensors Journal*, vol. 17, no. 13, pp. 3949–3960, July 2017.

[13] R. C. Luo, C.-C. Yih, and K. L. Su, “Multisensor fusion and integration: approaches, applications, and future research directions,” *IEEE Sensors Journal*, vol. 2, no. 2, pp. 107–119, Apr 2002.

[14] N. Siddiqui and R. H. M. Chan, “A wearable hand gesture recognition device based on acoustic measurements at wrist,” in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, July 2017, pp. 4443–4446.

[15] S. Kang and J. W. Yoon, “Classification of home appliance by using probabilistic knn with sensor data,” in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, Dec 2016, pp. 1–5.

[16] S. C. Mukhopadhyay, “Wearable sensors for human activity monitoring: A review,” *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1321–1330, March 2015.

[17] P. Hart, “The condensed nearest neighbor rule (corresp.),” *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 515–516, May 1968.

[18] P. Rosero-Montalvo, P. Diaz, J. A. Salazar-Castro, D. F. Peña-Unigarro, A. J. Anaya-Isaza, J. C. Alvarado-Pérez, R. Therón, and D. H. Peluffo-Ordóñez, *Interactive Data Visualization Using Dimensionality Reduction and Similarity-Based Representations*. Springer International Publishing, 2017, pp. 334–342.

[19] S. Nunez-Godoy, V. Alvear-Puertas, S. Realpe-Godoy, E. Pujota-Cuascota, H. Farinango-Endara, I. Navarrete-Insuasti, F. Vaca-Chapi, P. Rosero-Montalvo, and D. H. Peluffo, “Human-sitting-pose detection using data classification and dimensionality reduction,” in *2016 IEEE Ecuador Technical Chapters Meeting (ETCM)*, Oct 2016, pp. 1–5.

[20] S. Ziegler, R. C. Woodward, H. H. C. Iu, and L. J. Borle, “Current sensing techniques: A review,” *IEEE Sensors Journal*, vol. 9, no. 4, pp. 354–376, April 2009.

[21] O. Costilla-Reyes, P. Scully, and K. B. Ozanyan, “Temporal pattern recognition in gait activities recorded with a footprint imaging sensor system,” *IEEE Sensors Journal*, vol. 16, no. 24, pp. 8815–8822, Dec 2016.

[22] P. Hart, “The condensed nearest neighbor rule (corresp.),” *IEEE Trans. Inf. Theor.*, vol. 14, no. 3, pp. 515–516, Sep. 2006. [Online]. Available: http://dx.doi.org/10.1109/TIT.1968.1054155

[23] S. Garcia, J. Derrac, J. Cano, and F. Herrera, “Prototype selection for nearest neighbor classification: Taxonomy and empirical study,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 417–435, March 2012.

[24] P. Rosero-Montalvo, D. H. Peluffo-Ordez, A. Umaquinga, A. Anaya, J. Serrano, E. Rosero, C. Vsquez, and L. Suarz, “Prototype reduction algorithms comparison in nearest neighbor classification for sensor data: Empirical study,” in *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*, Oct 2017, pp. 1–5.

[25] J. A. Rodger and J. A. George, “Triple bottom line accounting for optimizing natural gas sustainability: A statistical linear programming fuzzy ilowa optimized sustainment model approach to reducing supply chain global cybersecurity vulnerability through information and communications technology,” *Journal of Cleaner Production*, vol. 142, no. Part 4, pp. 1931 – 1949, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959652616319345

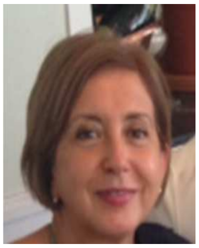
[26] R. Hu and J. Xia, “Calibration transfer of near infrared spectroscopy based on ds algorithm,” in *2011 International Conference on Electric Information and Control Engineering*, April 2011, pp. 3062–3065.

[27] D. F. Peña-Unigarro, J. A. Salazar-Castro, D. H.Peluffo-Ordóñez, P. D. Rosero-Montalvo, O. R. Oña-Rocha, A. A. Isaza, J. C. Alvarado-Perez, and R. Theron, “Interactive visualization methodology of high-dimensional data with a color-based model for dimensionality reduction,” in *2016 XXI Symposium on Signal Processing, Images and Artificial Vision (STSIVA)*, Aug 2016, pp. 1–7.

[28] S. G. Devi, K. Selvam, and S. P. Rajagopalan, “An abstract to calculate big o factors of time and space complexity of machine code,” in *International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2011)*, July 2011, pp. 844–847.

Paul Rosero-Montalvo He received his Master’s degree from Universidad de las Fuerzas Armadas ESPE, Ecuador, in 2016. Currently, he lecturer at the Faculty of Engineering in Applied Science at the Universidad Tecnica del Norte - Ecuador. His teaching experience is manly on basic electronics, microprocessors and embedded systems. He works extensively on Data Mining, intelligent systems and machine learning algorithms applied to the desig of Wireless Sensor Networks and Embedded Systems. He has published Articles in Data visualization, Business Intelligence, IoT and Sensors.

Diego Hernn Peluffo Ordez was born in Pasto - Colombia in 1986. He received his degree in electronic engineering, the M.Eng. and PhD degree in industrial automation from the Universidad Nacional de Colombia, Manizales - Colombia, in 2008, 2010 and 2013, respectively. He undertook his doctoral internship at KU Leuven - Belgium. Afterwards, he worked as a post-doc at Universit Catholique de Louvain at Louvain la-Neuve, Belgium. Currently, he is working as a researcher/professor at Yachay Tech - Ecuador. He is supervisor and external member of ALEPHSYS (Algorithms embedded in Physical Systems) research group from Universitat Rovira i Virgila - Spain. He is invited lecturer at Universidad de Nariño - Colombia and Corporación Universitaria Autónoma de Nariño - Colombia. He is the head of the SDAS research group (www.sdas-group.com). His main research interests are dimensionality reduction, and spectral methods for data clustering and representation.



Vivian Félix López Batista She Received a PhD. in Computer Science from the University of Valladolid in 1996. Since 1998, she is an Associate Professor of Computer Science at the University of Salamanca (Spain). She is member of the Data Mining Group (http://mida.usal.es/) and the Bisite Group (http://bisite.usal.es/) at the University of Salamanca (Spain). She has done research on natural language processing and neural networks. She has also 70 papers published in recognized journals, workshops and symposiums, 16 books and book chapters, and 20 technical reports. She has been member of the organising and scientific committee of several international symposiums such as IWPAAMS, PAAMS, CEDI.

Jorge Serrano He has a solid background in materials physics and spectroscopy but in parallel has a background in education, emotional counseling and leadership. He graduated in Physical Sciences at the University of Valladolid, Spain, had the opportunity to work and to obtain his doctorate in the Max Planck Institute in Stuttgart. It was his opportunity to learn what it means to work in a laboratory, find autonomy and learn permanently. After obtaining his doctorate he went to do a postdoctoral course at the cole Polytechnique in France. He then moved to Grenoble, France, where he participated in an international center dedicated to producing X-rays that can be used by different scientists to conduct experiments. In addition to his own research projects, he was there to advise researchers around the world for three years.

Edwin Rosero Rosero He is an ecuadorian researcher. He received his master degree in university research/teaching from Universidad Nacional de Loja - Ecuador, and in Business Intelligence from Universidad de los Andes - Ecuador. He has an experience of over 30 years in university teaching, in addition to administrative staff experince as director of the program of industrial engineering, dean of the faculty of engineering in applied sciences at Universidad Técnica del Norte. His research experience is mainly focused on wearables and conductive thread.

remote sensing

MDPI

Letter

Intelligent WSN System for Water Quality Analysis Using Machine Learning Algorithms: A Case Study (Tahuando River from Ecuador)

Paul D. Rosero-Montalvo [1,2,*], Vivian F. López-Batista [1], Jaime A. Riascos [3,4] and Diego H. Peluffo-Ordóñez [4,5,6]

[1] Department of Computer Science and Automatics Salamanca, Universidad de Salamanca, 37008 Salamanca, Spain; vivian@usal.es

[2] Department of Applied Sciences, Universidad Técnica del Norte, 100150 Ibarra, Ecuador

[3] Department of Engineering, Universidad Mariana, 520001 Pasto, Colombia; jandresr@umariana.edu.co

[4] Department of Engineering, Corporación Universitaria Autónoma de Nariño, 520002 Pasto, Colombia; dpeluffo@yachaytech.edu.ec

[5] School of Mathematical and Computational Sciences, Universidad Yachay Tech, 100650 Urcuquí, Ecuador

[6] SDAS Researh Group, 100150 Ibarra, Ecuador

[*] Correspondence: pdrosero@utn.edu.ec

Received: 24 April 2020; Accepted: 12 June 2020; Published: 20 June 2020

Abstract: This work presents a wireless sensor network (WSN) system able to determine the water quality of rivers. Particularly, we consider the Tahuando River from Ibarra, Ecuador, as a case study. The main goal of this research is to determine the river's status throughout its route, by generating data reports into an interactive user interface. To this end, we use an array of sensors collecting several measures such as: turbidity, temperature, water quality, pH, and temperature. Subsequently, from the information collected on an Internet-of-Things (IoT) server, we develop a data analysis scheme with both data representation and supervised classification. As an important result, our system outputs a map that shows the contamination levels of the river at different regions. Furthermore, in terms of data analysis performance, the proposed system reduces the data matrix by 97% from its original size, while it reaches a classification performance over 90%. Furthermore, as an additional remarkable result, we here introduce the so-called quantitative metric of balance (QMB), which measures the balance or ratio between performance and power consumption.

Keywords: prototype selection; river pollution; supervised classification; WSN

1. Introduction

Rivers are natural watercourses that commonly come from both precipitation (surface runoff), and snowpacks (e.g., water stored in glaciers). Regularly, they flow towards lakes, sea, oceans, or another river. Urban rivers are responsible for providing water resources to crops and human beings as well as navigation purposes. Certainly, this natural resource may not be everlasting. As a matter of fact, there is currently a great deficit of water reserves due to deforestation, inappropriate and excessive use of fertilizers and pesticides, causing environmental issues [1,2]. Likewise, the urbanization and industries have had collateral adverse impact directly on the water quality of river ecosystems worldwide [3]. Besides, the population growth produces enormous wastewater that enters into the rivers without any environmental control. United Nations (UN) settled that 90% of such waste is not correctly treated, and 70% of the industries discharge contaminant content without any adequate standards or rigorous inspections [4,5]. Water pollution contains high levels of biochemical oxygen demand (BOD), nitrogen, and phosphorus. So it is necessary to develop systems that support the detection and measuring of the contamination levels in rivers to maintain an optimal ecological balance, limiting environmental damage and preventing diseases spread [6]. Consequently, city governments have stated environmental policies intended to create urban regeneration initiatives around the care of their rivers [7]. In this connection, Ecuador, as our case study, has no any short or long-term plan to improve either the urban or rural river conditions [8].

Traditionally, water quality monitoring uses collected samples for laboratory testing, enabling then a wide range of analyses. Notwithstanding, it results impractical to manually measure water pollution at different points along the river. Moreover, this sort of tests may take a few days, and probably not reaching as a good precision as that of in-situ sampling [9]. Nowadays, the use of sensors for monitoring environmental conditions has received significant attention due to the real-time data collection, flexibility, and portability [4,10]. Following from this, the creation of a wireless sensor network (WSN) that combines several sensors with a data processing system and wireless communication can allow for an adequate measure of the water quality, where each sensor becomes a node that shares information among them as well as to a central server [11,12]. Thus, these data are greatly useful for further robust analyzes of water pollution in rivers. However, the large amount of data demands the implementation of machine learning algorithms to create systems that automatically can detect high levels of water pollution and make proper decisions. For that purpose, historical data (training data) become valuable to turn WSN nodes into intelligent systems [13,14].

Consequently, this work presents a novel system composed of three WSN nodes for monitoring in real-time the water pollution present in the Tahuando River (located in Ibarra, Ecuador) using machine learning algorithms. To do so, we establish different measurement points wherein each WSN node acquires the river's conditions data to be later processed internally by the system. In this sense, we consider water-quality variables, namely pH, turbidity, temperature and dissolved solids. Additionally, we carry out a sensor integration and calibration stage for eliminating reading errors. Finally, we sent these data to a cloud server, using a mobile network, where we visualize the node's information with its proper geo-location. As relevant results, a reduction of the required training set of 97% is accomplished by using is the condensed nearest neighbor (CNN) method as a prototype selection approach, as well as the classification stage—with k-NN—reaches 90.6% of performance. Then, our work is an exploratory study on different methods for both prototype selection and data classification applied to water treatment. Therefore, we have no gold standard result or benchmark method. Instead, an exhaustive comparison of representative methods is presented.

The fact that the data analysis process is implemented directly into the WSN represents a novelty itself for the development of both intelligent embedded systems, and data analysis platforms under low-computational resources. The rationale of creating an intelligent system including in-situ data analysis tasls (e.g., data classification) lies in the fact that an embedded systems can perform automatic decision-making processes with no requiring an external server. As well, it enables the possibility that even non-expert operators can readily interact with the system. In addition, it represents a solution to one of the main open issues of WSNs design, namely: information redundancy, which constraints the battery life-time, and often requires the incorporation of an external server for decision-making procedures. Additionally, to display a report of the current river's status, we implement an interactive user interface.

The rest of the manuscript is organized as follows: Section 2 gathers some remarkable related works. Section 3 describes both the system design and the data analysis proposed for implementing the machine learning algorithms. Section 4 presents the tests and results. Finally, Section 5 gathers the concluding remarks.

2. Related Works

Some works [5,6,9,15] have extensively worked on the estimation of water pollution, presenting different solutions for determining pollution state and its levels along several rivers located in China. Other works [16,17] analyze river status using satellite photographs. Meanwhile, in [4,10] WSN are instead preferred for data acquisition.

The work presented in [18] develops a WSN to determine the water quality level for human consumption through GPRS-generated data analysis, which is carried out on an external-to-WSN server holding a communication module. Similarly, another work [19] uses a high-performance external server. Specifically, it presents a system able to measure the quality of the water stored in tanks or reservoirs. In this connection, other works have proposed alternatives to improve the data processing aimed at reaching an admissible performance while involving a lower computational burden. An approach to do so is by minimizing the communication load, as done in [20] wherein an additional data compression stage is incorporated—particularly, the principal component analysis (PCA) algorithm is used. By compressing (or reducing the dimensionality of) data, the sending-packets process through WSN is enhanced in terms of performance and processing time. Similarly, the work presented in [21] performs a data analysis including temperature, pH, electrical conductivity (EC) and dissolved oxygen (DO) sensors, whose data are processed on a server and its result is sent back to the proposed WSN for decision-making. Another approach, which is becoming a new embedded systems paradigm is the design of intelligent systems performing an in-situ data analysis. For instance, in [22] the redundancy is minimized following a data fusion criterion to better manage the WSN computational resources, and bring an adequate energy consumption. Under this new paradigm where data analysis is carried out into the same system handling the data acquisition, the design of a system related to water quality monitoring results not only novel but proper. Indeed, on doing so, there would be enabled an affordable, large-coverage and easy-to-use WSN system, which along with right sensors will help environmental or health-related agencies or bodies to effectively make decisions regarding the quality of natural water from a specific source. Following from this, the work [22] involves stages for data acquisition, processing, and visualization.

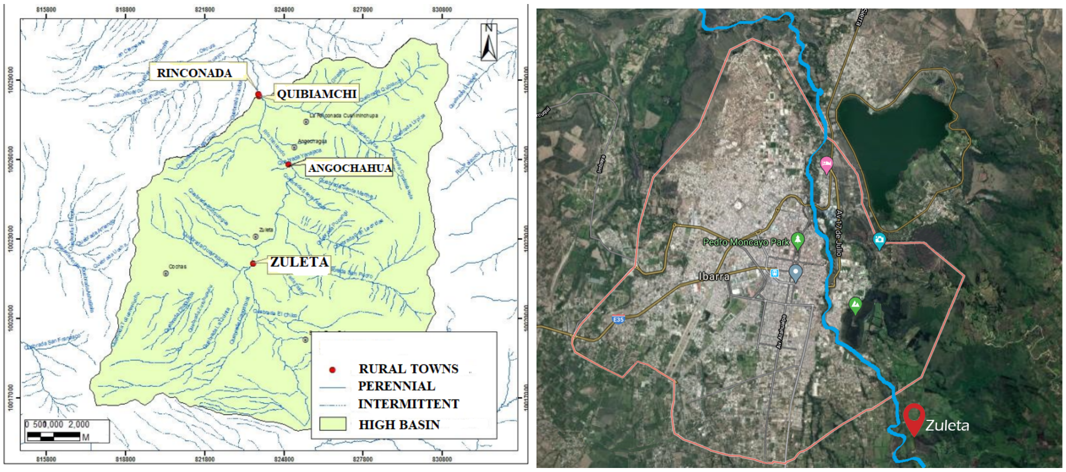
Nonetheless, no one of these solutions presents an in-situ data analysis. From the reviewed literature, only [23] presents an analysis of rivers in Ecuador. All of the aforementioned works presented appealing solutions to determine the water conditions of different rivers. However, in spite of all these efforts, there are still many open issues, such as: real-time data analysis, sensor calibration, and sending information to storage servers located far away from the acquisition point, among others.

3. Materials and Methods

Broadly, the proposed system consists of the following stages: initial conditions of the study region (Section 3.1), WSN design for accurate data acquisition (Section 3.2), and the data analysis with both the criteria for prototype selection, and supervised classification (Section 3.3).

3.1. Initial Conditions of the Study Region

The city of Ibarra (Ecuador) is the capital of the province of Imbabura with a dry-temperate climate of 18 °C on average. The urban population is 109 thousand and a rural population of approximately 45 thousand inhabitants. Its main commercial activity is the production of wooden articles and services to medium-scale industries. Regarding its water supply, 90% is carried out through the public distribution network, while the rest is for the use of river and vertier water [24]. Tahuando river is an important water resource in the Imbabura province, being part of the natural system of Ecuador. Due to its ability to transport and the flowing of its waters, it can withstand a large number of pollutants. However, there are several modifications at the ecological level, such as the loss of aquatic species, foul-smelling, and watercolor changes, among others. In Ecuador, only 10% of wastewater is treated. In Ibarra, around 600 liters per second of these waters are discharged into the Tahuando River, causing that no urban regeneration based on the increase in tourism can be carried out [25]. The Tahuando River is located at 0.4° latitude and 78.13° longitude. It encompasses an extension of 12 km from the community of Pesillo towards Salinas, in the Ibarra city. Figure 1 depicts the geographical location and basin.



(**a**) Rural section (**b**) Urban sector

Figure 1. Geographical description of Tahuando river. Zoomed view of the river route highlighting remarkable surrounding communities in Ibarra city's urban sector (**right**), and a widespread view regarding the rural section of Imbabura province (**left**).

3.2. Wireless-Sensor-Network Design

The design of our WSN approach is followed from the considered water-quality-related variables: pH, turbidity, temperature and dissolved solids. The considered sensor network is as follows: Firstly, we measure the turbidity and identify what kind of pollutants can be found in the river, such as: wastewater, chemicals, among others. Secondly, we use a pH sensor to determine if the water composition is acidic or basic as well as a quality sensor (total dissolved solids, TDS) to assess the level of dissolved oxygen in the water (cleanliness) [9]. Thirdly, we incorporate a temperature sensor to determine the water's changes and its relationship with the rest of the variables. To suitably develop the WSN network, we consider several operational requirements in the selection of sensors, such as reliability, precision, availability, ease-of-use, and scalability. Furthermore, in the selection of the WSN network processor system, we consider the number of pins and sensor libraries, as done in a previous work [11]. Specifically, the considered sensors are: `SKU: SEN0189` (turbidity), `SKU: PH-7BNC` (pH), `Ds18b20` (temperature), `RB-Dfr-797` (TDS). As well, the `Arduino Uno` is selected as processing system. Additionally, we use both global position module (GPS) and mobile communications (GSM) `Sim808` to send data. Finally, there is a `Lipo rider` battery manager for power supply with a solar charging system. Figure 2 presents the considered sensors along with the processor system (`Arduino Uno`).

Likewise, we calibrate each sensor as follows: sensor `SKU:PH-7BNC` (pH) has a linear response, so its tuning is based on measuring the voltage of several pH solutions. Particularly, we use two solutions, the first one was pH = 4.01, getting a voltage of 2.98 volts; meantime, the second one was pH = 6.86, obtaining a voltage of 2.53 v. Thus, the equation to obtain the estimated pH is:

$$\mathrm{pH} = -5.65 * (v_1) + 21.15, \tag{1}$$

with v_1 as the voltage obtained by the sensor `SKU:PH-7BNC`. Likewise, the turbidity sensor `SKU: SEN0189` gives a reading ranging between 2.5 to 4.3 volts with values between 3000 and 0 turbidities (NTU), respectively. According to its datasheets, we can write the following equation:

$$\mathrm{NTU} = -1120.4 * {v_2}^2 + 5742.3 * v_2 - 4352.9, \tag{2}$$

where v_2 is the voltage registered by the sensor SKU:SEN0189. On the other hand, the datasheet of the temperature sensor Ds18b20 indicates that each Celsius degree can be transformed using the equality 10 mv = 1 °C; thus, the equation is:

$$Temp = \frac{v_3 * 5}{1023 * 0.01}, \tag{3}$$

with v_3 as the voltage obtained by the sensor Ds18b20.

Finally, the TDS RB-Dfr-797 sensor provides a flexible calibration protocol, with a reset button, we can return to the initial conditions, that is, a TDS value of 23 mv. Consequently, we refresh the Arduino program and use the next equation:

$$\text{TDS} = \frac{(30 * 5 * 1000) - (75 * v_4) * 5 * (1000/1024)}{75 - 0.23}, \tag{4}$$

where v_4 is the voltage obtained by the sensor RB-Dfr-797.

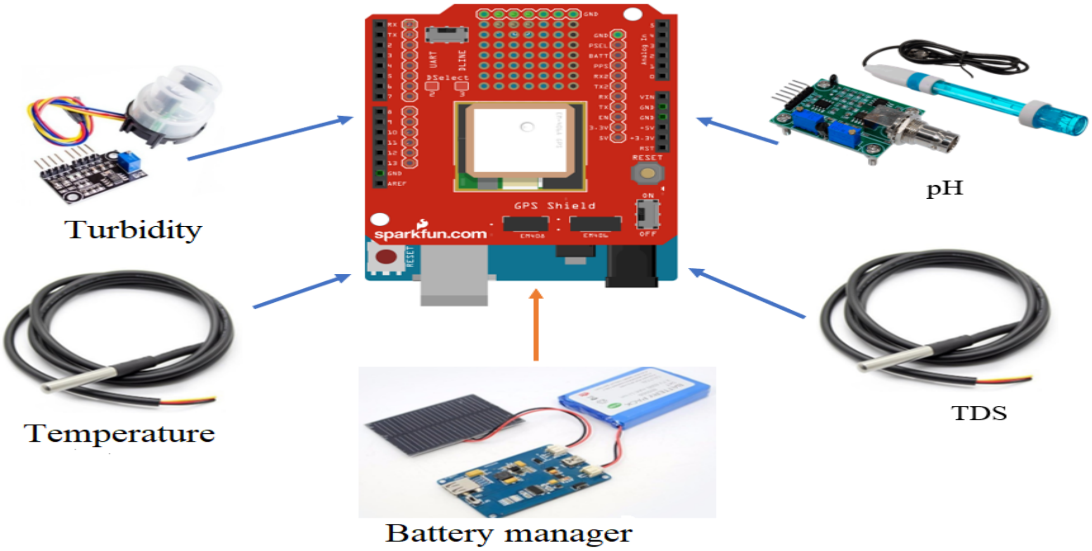


Figure 2. Demonstrative diagram of the proposed WSN system. Considered sensors (SKU: SEN0189 (turbidity), SKU: PH-7BNC (pH), Ds18b20 (temperature), RB-Dfr-797 (TDS)), and the processor (Arduino Uno).

Upon sensor configuration, each v_i value will correspond to a digital-analog converter (DAC) with a resolution of 10 bits, already in the microprocessor Arduino Uno. Furthermore, we implement the moving average recursive filter to reduce the acquisition errors and smoothing the signal from each DAC. This filter takes a subset (window) of N samples, and calculate its arithmetic average to estimate a filtered sample as [26]. This filter is implemented in each DAC separately through the following equation:

$$y_n = (2n+1)^{-1} \sum_{i=n-d}^{n+d} x_i, \tag{5}$$

where $\mathbf{x} = (x_1, \ldots, x_{L_x})$ is the input signal, $\mathbf{y} = (y_1, \ldots, y_{L_y})$ is the filtered signal, d is the window size, and L_x and L_y are respectively the input and filtered signal lengths. To accounting for a reduction of the computational resources usage, we experimentally define $d = 11$.

With the aim of verifying the data obtained by each sensor and validating the reliability thereof, samples obtained from the river are taken to the Environment Services Laboratory of the Technical

University of the North (Universidad Técnica del Norte (Universidad Técnica del Norte official web site: https://www.utn.edu.ec/web/uniportal/) from Ibarra-Ecuador, as they count on the technology and reagents to make comparison against the data obtained by the WSN. In this sense, following reliability criteria for each sensor, some recommended performance measures are considered, such as: (i) Accuracy: ability to provide the same reading by repeatedly performing the same experiment (standard deviation), (ii) Reproducibility: ability to reproduce the same results when modifying initial conditions of the experiment, and (iii) Stability: ability to produce the same output value in a long time. Overall obtained results are gathered in Table 1, which correspond to 10 tests over controlled environments to assess the data stability. As can be appreciated, the collected data from the sensors exhibit an error average of 5% in contrast to the those generated at the laboratory—such an error is acceptable enough for implementation purposes.

Table 1. Sensor performance metrics.

	Sensors			
Measure	`SEN0189` (turbidity)	`PH-7BNC` (pH)	`Ds18b20` (temperature)	`RB-Dfr-797` (TDS)
Precision	7 ±	3 ±	5 ±	5 ±
Reproducibility	It is necessary to wait up 2 s for calibration to be done	Adequate	Adequate	Some reading errors
Stability	Adequate	3 ±, variable for each test	Adequate	Adequate

3.3. Data Analysis Paradigm

For a proper and wide data acquisition, we establish three node points in different locations, based on the population density of Ibarra, as follows: (i) *La Rinconada*, with low population and located at the river's beginning; (ii) *El Tejar*, with middle population rate and some wastewater discharged into the river; and (iii) *La Victoria*, with a larger population density and more discharge of pollutants from the city. Figure 3 shows the geographic locations of the nodes. Furthermore, we label each data from the nodes with a localization tag. For the data acquisition procedure, we design a collection protocol as follows: A schedule consisting in four collecting times is set, namely: in the morning, afternoon, night and early morning. Such a schedule is timed with `Timer2`, which is an Arduino internal clock. So, the system is timed for alerts at 08:00, 12:00, 17:00 and 00:00. On those times, the system records the sensor readings every 10 min for two hours (amounting to 6 samples per hour). Finally, these captured data are sent to the remote server through the GSM/GPS sensor. This collection protocol was performed during 3 months, generating an enough amount of information to be used in the subsequent data analysis stage.

Once the data are stored in an external server, a two-stages data analysis process is carried out: The first stage is the training set size reduction—via prototype selection—involving the least or no affectation to the intrinsic knowledge they hold. The second one is the classification task, in which the the algorithm that best fits the first stage while keeping a high accuracy is sought. Both stages are set and performed under low-computational cost criteria (given the device conditions). This process is carried out in order to be compiled within each WSN node (including both prototype selection, and classification). Then, system is able to make their own decisions based upon the reduced, stored dataset as well as the implemented classification algorithm. Therefore, on the one hand, the adaptability criterion required by an intelligent system is met, by making it able to be used anywhere on the river. On the other hand, the resulting system requires no re-run the data analysis process and thus it can be readily used by any system operator whom is not required to hold an expertise on embedded systems or data analysis, but only knowledge in water treatment itself.

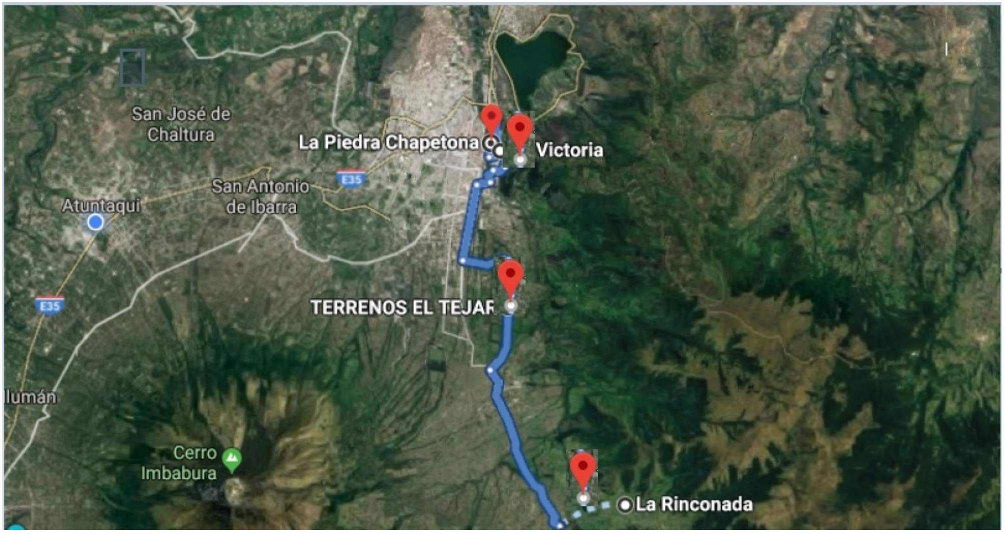


Figure 3. Geographic location of the WSN nodes. Spots strategically selected to acquire data from, in order to encompass representative zones, as well as different types and levels of river pollution.

3.3.1. Proposed Quality Measure: Quantitative Metric of Balance (QMB)

Algorithm analysis is an important part of designing thereof. Traditionally, the analysis of programming code or algorithms lies in applying theoretical and mathematical procedures. Indeed, when selecting supervised classification algorithms, efficient programs must be ensured to be created, as this translates into better power consumption and therefore battery life time usage. In this sense, the here-introduced Quantitative metric of balance (QMB) is aimed at quantifying how proper is the ratio between classifier performance and the data size reduction by the prototype selection stage. In this connection, the closer its value is to 100%, the better the ratio. As these three individual measures have an increasing nature, we multiply them to state a single value, namely, the rate of removed instances (RI) times the classification performance (CP), and divided by the response time of the classification algorithms (RT), as follows:

$$\text{QMB} = \frac{(RI * CP)}{RT} * 100\%. \tag{6}$$

Certainly, some classification criteria make use of mathematical functions or recursive functions of model adjustment that, when coded in a low-level language (assembler), generate response time delays, memory saturation and an excessive battery consumption. In this sense, the proposed QMB is aimed at penalizing the excessive computational cost in order to make it more feasible the implementation of data analysis algorithms into an embedded system. Besides, since it takes into consideration the number of removed training set instances to quantify the overall performance, this metric rewards the classification algorithm if it requires the least memory capacity when performing the decision-making procedures. When operating under real conditions, the system acquires the data from the sensors, filter the acquisition errors, make the decision through its compiled classification algorithm, and use the selection of prototypes to determine if this new reading improves the prediction ability of the system. If so, it is added into the training matrix otherwise it is only sent to the external server for visualization purposes.

Figure 4 shows the proposed data analysis scheme.

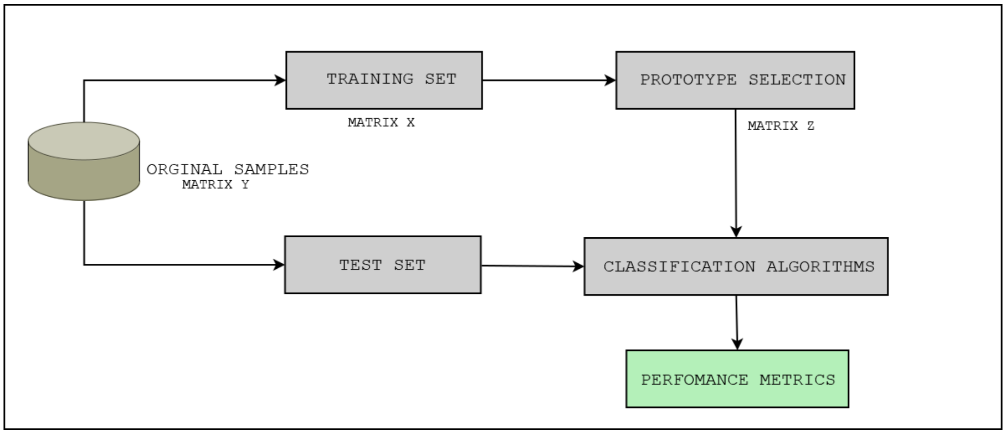


Figure 4. Data analysis scheme including prototype selection and classification stages.

3.3.2. Prototype Selection

Since WSN systems have limited computational resources, its battery consumption is directly related to the amount of data to be processed, and therefore the implementation of machine learning algorithms into thereof is limited. In this connection, the prototype selection (PS) techniques may take place by reducing the training matrix size, while utmost maintaining as good classification performance as that obtained when considering the original size. Regarding PS algorithm designing, technical literature reports at least three main methods (namely, compensation-based, edition-based, and hybrid) [27]. As have been mentioned throughout this paper, the whole process is carried out in such manner that the prototype selection results (reduced data matrices) can be stored directly into every WSN node.

In this work, in order to account for an enough coverage, we have chosen three representative algorithms of each method, as follows:

- `Condensation:` Condensed Nearest Neighbor (CNN), Reduced Nearest Neighbor (RNN), and Selective Nearest Neighbor (SNN).
- `Edition:` Edited Nearest Neighbor (ENN), All-k Edited Nearest Neighbors (AENN), and Iterative Partitioning Filter (IPF).
- `Hybrid:` Decremental Reduction Optimization Procedures 2 (DROP 2), Decremental Reduction Optimization Procedures 3 (DROP3), and Iterative Noise Filter based on the Fusion of Classifiers (INFFC).

3.3.3. Classification Algorithms

Classification algorithms can learn based on different criteria, having each of them representative algorithms [27]. Herein, we consider four criteria and their respective representative algorithm, namely:

- Distance-based: K-Nearest Neighbors (KNN).
- Model-based: Support Vector Machine (SVM).
- Density-based: Bayesian classifier (BC).
- Heuristic: Decision Tree (DT).

Given that the four aforementioned criteria are essentially different, a comparison of individual performances is necessary to identify the one(s) best fitting the nature of data and classification task. As well, it is of crucial interest to measuring the computational cost that each algorithm involves to be further implemented within the WSN node.

The database—obtained according to the pollution level—has been divided regarding the information acquired by the WSN nodes into 3 types (being our training labels): high, medium and low contamination. Therefore, if the system is located at different spots along the river, it can generate a map of the pollution status and estimate the river's course. Alternatively, if it is located statically, the system can determine, in hours, how the level of contamination varies with respect to the time of day.

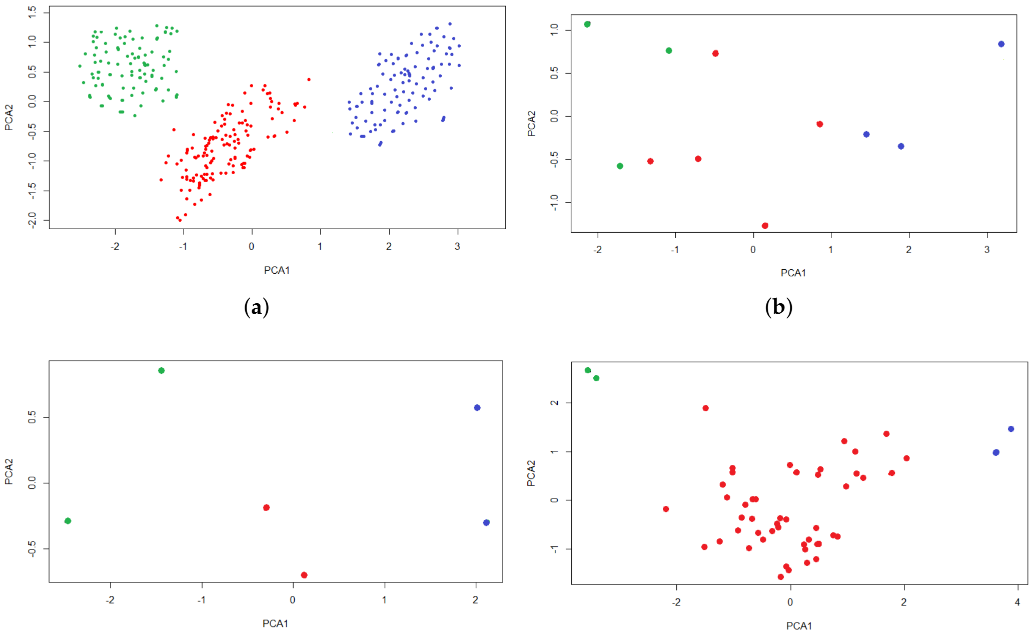
4. Results and Discussion

In order to evaluate the behavior of each stage, we firstly discuss the data reduction in the training matrix. Subsequently, we show the outcome of our proposed analysis scheme, namely, the performance analysis using our defined metric (QMB) for determining the ideal algorithms for its implementation in the WSN nodes. Finally, we present the results of the final implementation of the system and the tests in real environments.

4.1. Data Reduction

The sensors were acquiring data during the months of July, August and September on random days. As a result, we obtained the data matrix called $\mathbf{Y} \in \mathbb{R}^{m \times n}$, where m is the number of instances, and n the number of measured variables (sensors). While, $\mathbf{L} \in \mathbb{R}^{m \times 1}$ is the tag vector. Thus, we have that $m = 507$, and $n = 4$. With these data, we implemented the PS algorithms in order to reduce the training matrix and processing time. In addition, to validate the classification criteria, we retained 20% of the $\mathbf{Y}$ matrix for performance testing. In succession, the matrix for the data scheme is $\mathbf{X} \in \mathbb{R}^{p \times n}$, where $p = 405$. Table 2 shows the summary of the PS algorithms results and find a new reduced data matrix $\mathbf{Z}$.

Accordingly, we have selected the CNN, DROP1 and DROP3 algorithms as they reach the highest percentages of reduction in the database. Figure 5 shows scatter plots of the initial data set and the reduced versions generated by CNN, DROP1 and DROP3.



(**a**) (**b**)

(**c**) (**d**)

Figure 5. 2D scatter plots of resulting data matrices ***Z*** of the chosen prototype selection algorithms. (**a**) Data matrix ***X***, (**b**) CNN, (**c**) DROP1, (**d**) DROP3.

Table 2. Analysis of PS algorithms in relation to optimization embedded computational resources.

PS Algorithm	Exec. Time (s)	Remv. Inst	% of Remv. Inst
AENN	3.17	0	0
BBNR	125.23	102	25.18
CNN	2.28	394	97.28
DROP1	130.63	399	98.51
DROP2	230.28	354	87.407
DROP3	264.97	354	87.40
ENG	250	210	51.85
ENN	0.72	0	0
RNN	2.39	394	97.28

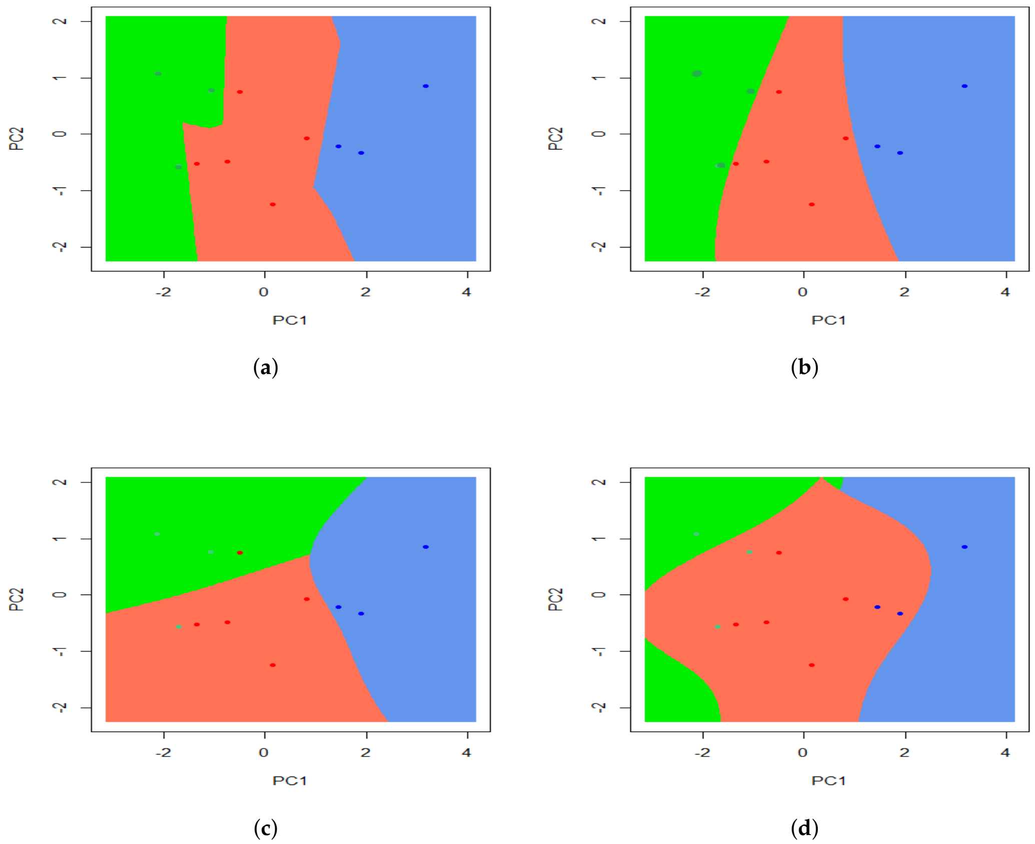
4.2. Classification Performance

With the reduced data sets, we compared the classification performance using the aforementioned algorithms. Table 3 summarizes the results of the classifiers with cross-validation with ten random folds.

Table 3. Classifier's metrics.

Classifier	Matrix X%	CNN%	DROP1%	DROP3%
Accuracy				
k-NN	97.6	90.6	93.6	95
Bayesian classifier	95	87.5	82.6	0.99
Decision Trees	99.3	66.9	33.33	33.33
SVM (Polynomial kernel)	100	75	75.3	92.14
SVM (Sigmoide kernel)	100	75	92	100
Sensitivity				
k-NN	96.6	88.3	91.6	93.3
Bayesian classifier	93.3	75	76	97.3
Decision Trees	99.3	33	33.33	33.33
SVM (Polynomial kernel)	100	94	66.9	92.14
SVM (Sigmoide kernel)	100	50	90	100
Specificity				
k-NN	98.2	93.6	95.3	96.6
Bayesian classifier	96.6	88.6	88	99.3
Decision Trees	99.6	66.9	33.33	33.33
SVM (Polynomial kernel)	100	97	89	100
SVM (Sigmoide kernel)	100	100	94	100
Precision				
k-NN	96.3.0	98.3	89.3	93.3
Bayesian classifier	93.3	100	66.6	98.3
Decision Trees	93.3	33.9	33.33	33.33
SVM (Polynomial kernel)	100	93	89	93.13
SVM (Sigmoid kernel)	100	50	86.6	100

To graphically appreciate the results of the whole data processing scheme, just as done in previous works [11,14], we use the principal component analysis conventional algorithm as a dimensionality reduction approach to represent the original data over a lower-dimensional domain. Figure 6 presents scatter plots regarding the two first principal components to depict the decision borders generated by every considered classifier. This process is carried out for demonstration purposes in order to know the algorithms' ability to differentiate each label in an understandable way for the human being perception (visual-type in this case).



(**a**) (**b**)

(**c**) (**d**)

Figure 6. Decision borders for each classifier. Original data are embedded into a bi-dimensional space using PCA to graphically depict the classification ability of the considered algorithms. (**a**) k-NN, (**b**) Bayesian classifier, (**c**) SVM (Sigmoid kernel), (**d**) SVM (Polynomial kernel).

Numerical results of the joint performance of the prototype selection and data classification are summarized in Table 4.

Table 4. QMB analysis for every classifier along with the previously identified PS algorithms.

Classification	**Exec. Time (s)**	**QMB Value**	
algorithm		CNN%	DROP1%
k-NN	1.21	72.85	76.20
Bayesian	1.85	46.01	43.97
Decision Tree	0.77	42.06	42.63
SVM (polynomial kernel)	5.2	14.03	14.2
SVM (sigmoide kernel)	6.1	11.96	12.11

Discussion on performance measures: As can be seen in the Table 3, VSM reaches the best classification performance based on the considered metrics (100%). Nonetheless, its algorithm involves mathematical functions (known as kernel functions), which are not able to readily processed in a WSN. In this connection, the proposed QBM allows for warning about this computational cost in relation to the amount of data used to train the classification algorithm and the system response time when assigning the corresponding label to a new data from the sensors. This can be appreciated from the fact that by reducing the training matrix its performance decreases significantly. The same occurs for all the considered algorithms excepting for k-NN, whose distance-based nature is non-expensive in terms of computational cost. Furthermore, by using a reduced data matrix, k-NN considerably maintains its performance. Furthermore, it is clearly noted that DROP1 is the best-suited algorithm for prototype selection although its computational cost is very high. Hence, given the design settings and the embedded systems conditions, CNN is preferred and therefore selected as the algorithm for prototype selection, while k-NN is considered as the selected classification algorithm reaching a performance of 90.6% and a QBM value of 72.85%.

4.3. Implementation and Testing

Figure 7 depicts the functional architecture of the nodes using the proper, selected prototype selection algorithms, which are to be compiled within thereof. As can be appreciated, each node holds the data-acquisition sensor set. The data analysis and processing is as follows: The raw data is first filtered by using the `Moving average` filter, which, in this case, is enough to remove the components (artifacts) related to reading errors and noise. Subsequently, data are classified by the algorithm `k-NN`, which assigns a label and decides about the predicted level of water contamination according to the training database and following a distance-based, majority-vote-driven approach. Then, data undergo an additional processing via `CNN` to determine whether the training database can be improved by removing instances exhibiting negligible relevance regarding either the subsequent classification task or the intrinsic knowledge they may hold. Finally, the output information is converted into a character string together with its label to be sent by the GSM network to the external server and display the data obtained from each sensor and the decision made. It is worth highlighting that the node to be monitored can be selected through the interface.

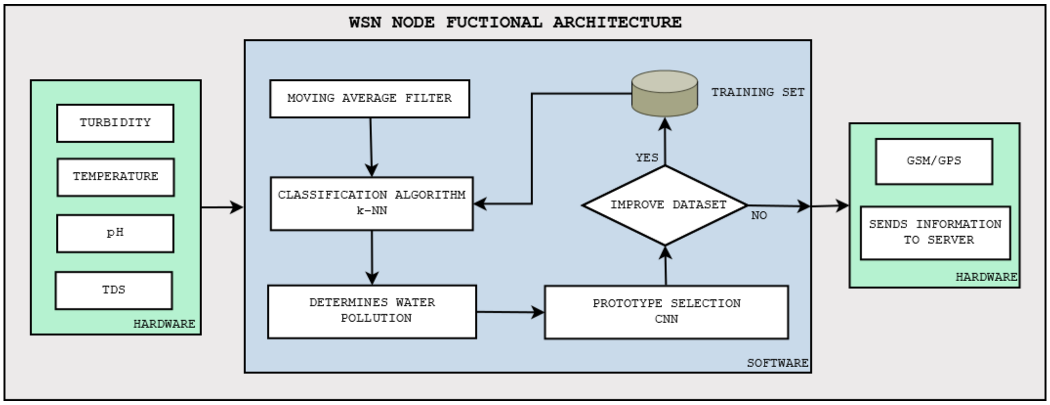


Figure 7. WSN node functional architecture incorporating the workflow of the in-situ data analysis and processing and mainly consisting in filtering, prototype selection and classification.

In the overall work-flow of our approach, the need for using an external sever lies in the fact that optimizing resource consumption at the in-situ analysis (directly on WSN Nodes) entails performing offline data processing tasks, mainly, at three specific points. The first one is when collecting data from each WSN node, being its main function the storing of such information (which—at this extent—corresponds to the outcomes of reading-errors-filtering stage produced by the moving average filter). The second one is the offline, exhaustive running, and comparison of classification algorithms to identify the ones reaching a good compromise between accuracy and computational cost, and therefore, being adequate to be directly implemented into the WSN nodes. Finally, as the third point, the server is used for information visualization purposes (displaying numerically and graphically the acquired data, the decision (classification) made by each node and the river pollution historical). This information is also stored in the server. Of course, those algorithms identified as adequate ones at the second point are the ones that are finally incorporated into the WSN nodes.

Once performed the data analysis procedures, we integrate all sensors into a PCB board incorporating an `Arduino Uno` as a processor unit. A view of the developed WSN node can be seen in Figure 8.

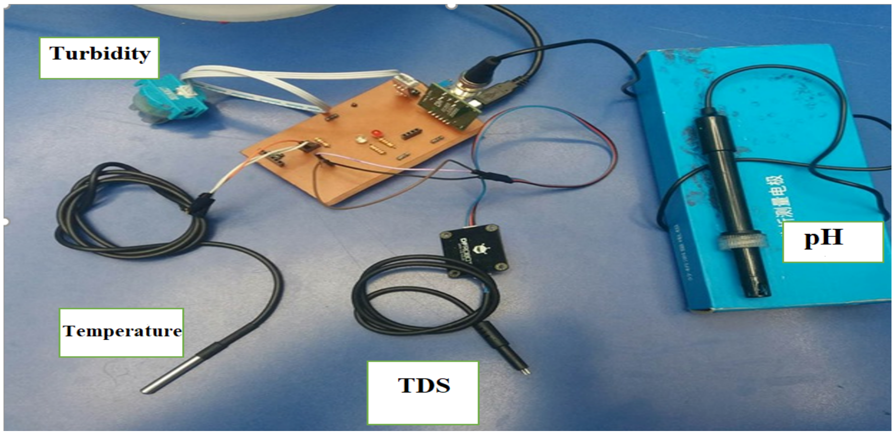


Figure 8. View of the WSN node including the four considered sensors and the processor.

The developed WSN has a considerably high operating consumption for a LiPo-type battery. To increase the life time of both the system and the battery, energy saving modes are used inside the Arduino board that handles the sensor activation. To enable such modes, we consider the use of timers, which work as an internal clock determining the data-acquisition-and-sending timing, and therefore limit current consumption. Hence the power consumption of every single sensor and the processor should be considered. In normal operation conditions, the total electric current consumption (considering all the sensors) amounts to 110 mA, while the GPS-GSM module and the Arduino require 40 mA and 45 mA, respectively. Meanwhile, when the battery saving system is enabled, the sensors and the GPS module are not is used, and thus only the Arduino works and is fed with 15mA. As stated in [28], the following equation relates the battery life time with the total power consumption (P):

$$P = \frac{(T_{on} * I_{on}) + (T_{sleep} * I_{sleep})}{T_{on} + T_{sleep}}, \tag{7}$$

where T_{on} , T_{sleep}, I_{on}, and I_{sleep} stand respectively for Normal Consumption Time, Sleep Consumption Time, Current Consumption at Normal Conditions, and Current Intensity Sleeping Consumption.

As explained in Section 3.3, the system is on during 10 min and then remains in battery saving mode. As a result, the system consumes 78.45 mA per hour. If the used battery is 5 volts at 1000 mA, the system can work continuously for 12.73 h. However, the system is activated only four times per day (early morning, mid-morning, afternoon and night), that is, it only works for 4 h a day. As a result, the system can remain for at least 3 days with no requiring battery manager support. As an advantageous aspect of our system we may say that, when implemented with a solar panel powering the battery, there is experimental evidence that it can work up to 4 months with no discharging or critical battery issues.

Subsequently, over the implemented system, we store the training dataset obtained after running the CNN algorithm, which is to denoted $\mathbf{Z} \in \mathbb{R}^{s \times n}$, by setting the number of prototypes as $s = 11$. At this extent, CNN algorithm is considered as an recommendable approach, since its execution time is the least while its ability to reduce the dataset instances is proper enough. Consequently, if the system requires to be reconfigured to train the classification algorithm model, the CNN algorithm can be compiled readily on the WSN network with no entailing extra battery consumption or diminishing the system performance. Then, we implemented the Bayesian classifier so that it can make system decisions concerning the tag assigned by location. Thus, we can determine the contamination levels (high, medium, low) using the nodes along the river. Since the system is intended to be waterproof, we use a river buoy to keep the system afloat. At its upper part, we install the solar panel and the GPS-GSM communication antenna. Furthermore, the nodes are anchored using an ironwork attached to the river stones, as shown in Figure 9.

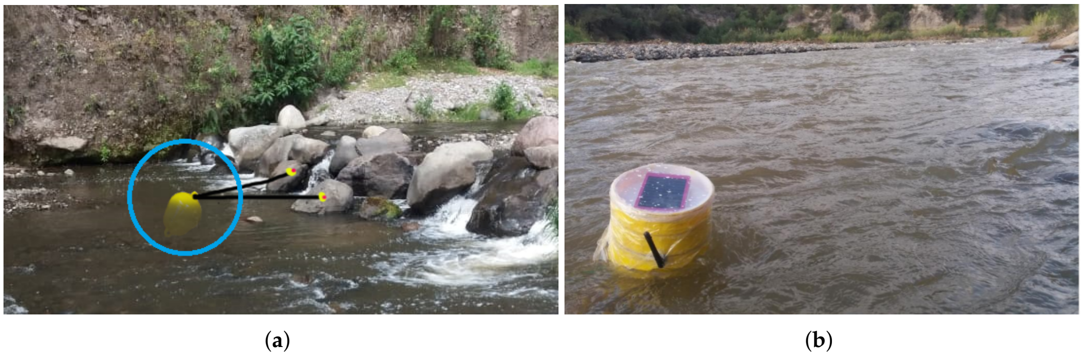


Figure 9. Anchored node acquiring and sending data to interface. (**a**) Simulation. (**b**) Real conditions.

Besides, for displaying purposes, we develop a monitoring interface in `Processing` using a local server that downloads and visualizes the information from the server. In this interface, we show the status of each sensor, the node location, and the level of contamination of the river. Figure 10 summarizes both the sensor testing and the visual interface with the decision taken.

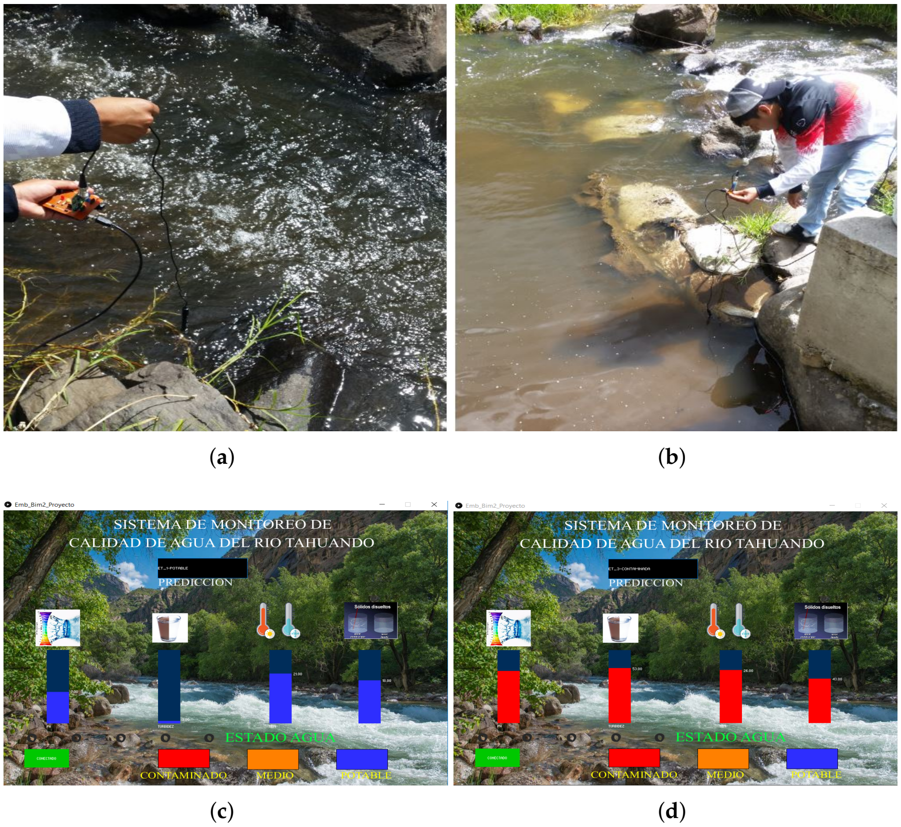


Figure 10. System testing and visual interface. (**a**) Testing embedded system developed in the rural sector. (**b**) Testing embedded system developed in urban sector. (**c**) Visual interface showing low level of contamination. (**d**) Visual interface showing high level of contamination.

For a more extensive analysis, we move the nodes throughout the river to assign a color label, based on the contamination level, as follows: red refers to high contamination, yellow to medium, and green to null or low pollution. Accordingly, Figure 11 shoes the contamination levels along the case-study river. As a relevant result, we identify that at the Campiña church zone there is already a high level of pollution.

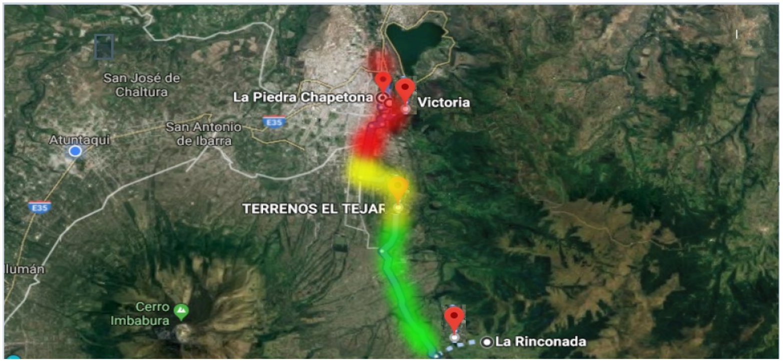


Figure 11. Tahuando river conditions along its stream bed.

Finally, with all nodes running, we daily capture data to observe the maximum values, in order to detect the hours of the day with highest contamination, which are in line with the human's work schedules. Figure 12 shows the pH, Temperature, and NTU values registered by the sensors during a whole day.

It is worth mentioning that our system may exhibit failures regarding the loss of signal from the GPS-GSM module when restarting it to carry out the data acquisition. To overcome this drawback, we follow a heuristic sensor calibration procedure as follows: On one hand, when activated, the system first turns on the GPS-GSM module so that there would be enough time to re-link to the GSM network and send back a status indicator signal. On the other hand, the length of the cables connected to the sensors was initially very long. This caused that when the volume of water decreased, cables descended to the bottom of the river and got brushed against stones. Consequently, since the length of the system-incorporated sensor is between 2 and 5 cm, an excessive wear on the sensors is induced. To cope with this issue, we search for and identify points where the river depth is the least possible varying, and is not prone to water stagnation.

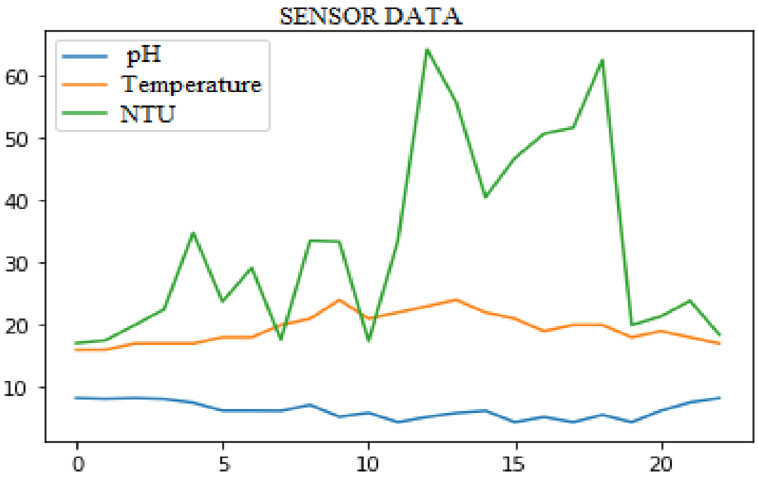


Figure 12. Sensor-generated data acquired per hour during a day.

5. Final Remarks

In this work, we present the complete design and validation of an intelligent wireless sensor network (WSN) system to measure the contamination levels of a river. Particularly, the Tahuando River is of interest. Broadly speaking, the proposed system involves two stages: electronic device implementation, and data analysis.

For the electronic design, since the case-study river may have high levels of pollution, as well as it may occur significant variations depending on the hours of the day, and zones of its route, we implement several WSN nodes for acquiring the river's conditions information by covering a meaningful zone and within a wide enough range of time. In this sense, we both calibrate and tune the sensors for a correct data collection. Additionally, we experimentally demonstrate that our data reading schedules were adequate for detecting higher pollution hours. Furthermore, we highlight that the river buoys is a key element to meet the node's permeability requirements as well as to enable the proper functioning of each WSN node.

Regarding the proposed data analysis scheme, we demonstrate that a classifier together with a prototype selection is suitable for a WSN-based water-quality monitoring system. It is reached a good trade-off between the computational resource usage (as the training matrix size is reduced to meet the system operation conditions), and the classification performance at detecting the pollution levels along the river. In addition, given the network coverage, the proposed system is able to send information from the WSN node to the server. Therefore, the filtered data can be visualized in an interface, and an in-situ analysis becomes possible. It is important to mention that the server is only for data visualization purposes and does not have the implementation of machine learning algorithms.

As a future work, the battery life is to be more carefully considered by exploring both different methods of extending its duration and alternatives sources of energy to supply the nodes (i.e., using the water flow to generate energy). A large number of nodes and wider coverage (located at different water resources around the province of Imbabura, Ecuador) is highly desirable for further In addition, we are intended to a seek for alternatives to mitigate system affectations due to disturbances caused by the presence of unexpected individuals (either people or animals), as so far our readily solution has been to locating the system in a hardly visible and difficult-to-access spot.

Author Contributions: P.D.R.-M.: Conceptualization, methodology, software, formal analysis, investigation, writing—original draft preparation, visualization, resources, V.F.L.-B.: investigation, supervision, project administration, J.A.R.: formal analysis, writing—review, visualization, D.H.P.-O.: validation, writing—review, methodology supervision, project administration. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work is supported by the Smart Data Analysis Systems Group—SDAS Research Group (http://sdas-group.com).

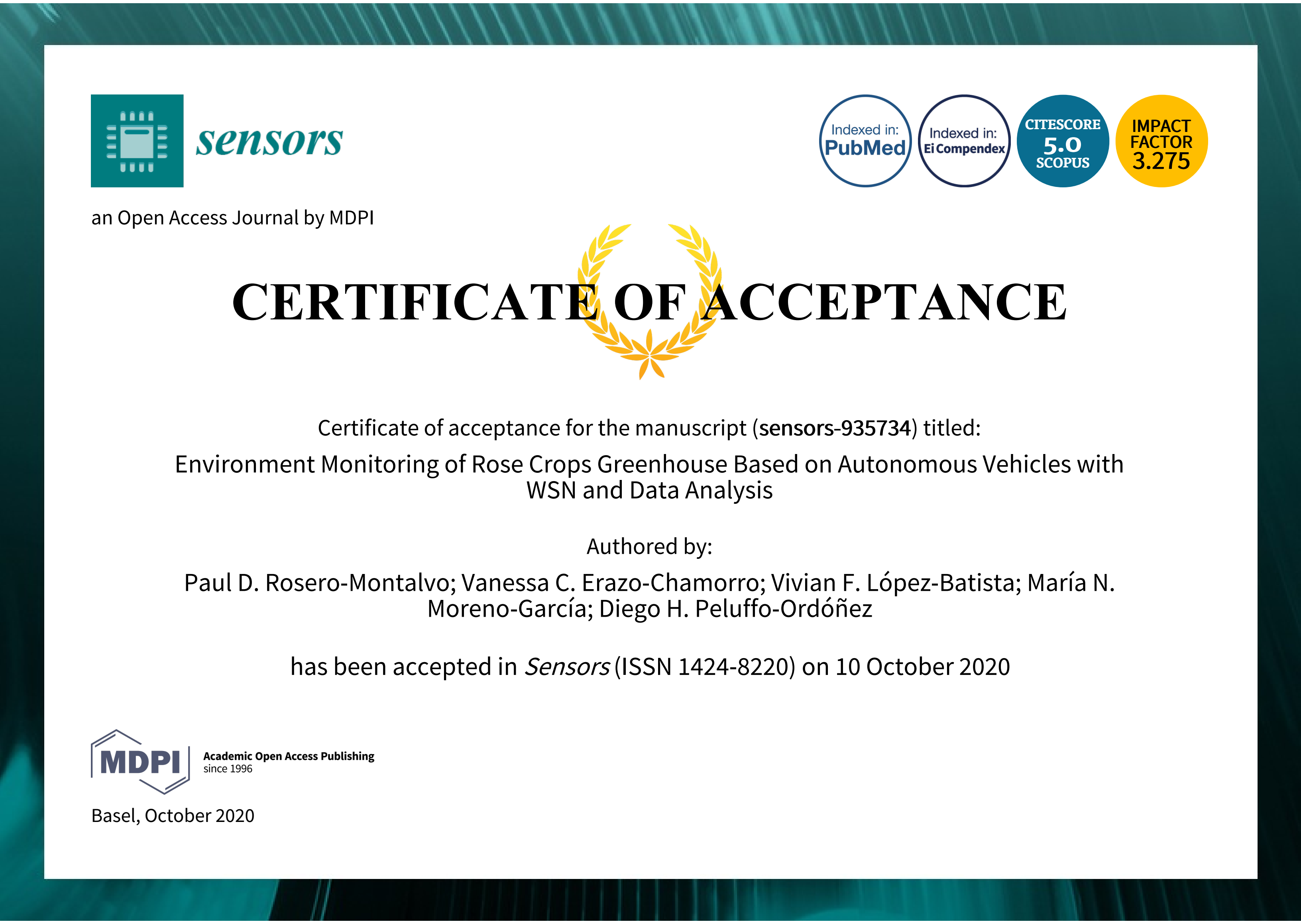
Conflicts of Interest: The authors declare no conflict of interest.

References

1. Venancio Cruz, D.; Rivelino Gomes de Oliveira, M.; Cunha Filho, M.; Venancio da Cruz, D. Monitoring pH with quality control based on Geostatistics Methodology. *IEEE Lat. Am. Trans.* **2016**, *14*, 4787–4791. [CrossRef]
2. Yang, C.; Wang, X. The water quality and pollution character in QingShuiHai lake valley-typical urban drinking water sources. In Proceedings of the 2011 International Conference on Remote Sensing, Environment and Transportation Engineering, Nanjing, China, 24–26 June 2011; pp. 7287–7291.
3. Zhang, Z.; Zhang, F.; Xu, C.; Xu, J.; Zhang, W.; Qi, Q. Study on the water environment capacity for the typical watershed in Taizihe River. In Proceedings of the 2011 International Symposium on Water Resource and Environmental Protection, Xi'an, China, 20–22 May 2011; Volume 1, pp. 486–488.

4. Randhawa, S.; Sandha, S.S.; Srivastava, B. A Multi-sensor Process for In-Situ Monitoring of Water Pollution in Rivers or Lakes for High-Resolution Quantitative and Qualitative Water Quality Data. In Proceedings of the 2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), Paris, France, 24–26 August 2016; pp. 122–129. [CrossRef]
5. Zhai, C.; Huang, Q.; Chang, J.; Gao, F. The study of water resources reasonable allocation of BaoJi area in Wei River with considering the ecology base flow. In Proceedings of the 2011 International Symposium on Water Resource and Environmental Protection, Xi'an, China, 20–22 May 2011; pp. 816–818. [CrossRef]
6. Guo, W.; Chen, J.; Sheng, Y.; Wang, J. Integrated evaluation of water quality and quantity of the Wei River reach in Shaanxi Province. In Proceedings of the 2011 International Symposium on Water Resource and Environmental Protection, Xi'an, China, 20–22 May 2011; pp. 863–866. [CrossRef]
7. Zhang, H.; Xie, X.; Hou, J. Water pollution accident control and urban safety water supply. In Proceedings of the 2011 2nd IEEE International Conference on Emergency Management and Management Sciences, Beijing, China, 8–10 August 2011; pp. 37–40.
8. De Agua, S. Biblioteca—Secretaría del Agua. Available online: https://www.agua.gob.ec/ (accessed on 1 January 2020).
9. Wang, J.; Guo, X.; Zhao, W.; Meng, X. Research on water environmental quality evaluation and characteristics analysis of TongHui River. In Proceedings of the 2011 International Symposium on Water Resource and Environmental Protection, Xi'an, China, 20–22 May 2011; pp. 1066–1069. [CrossRef]
10. Taufiqurrahman; Tamami, N.; Putra, D.A.; Harsono, T. Smart sensor device for detection of water quality as anticipation of disaster environment pollution. In Proceedings of the 2016 International Electronics Symposium (IES), Denpasar, Indonesia, 29–30 September 2016; pp. 87–92. [CrossRef]
11. Rosero-Montalvo, P.D.; Pijal-Rojas, J.; Vasquez-Ayala, C.; Maya, E.; Pupiales, C.; Suarez, L.; Benitez-Pereira, H.; Peluffo-Ordonez, D. Wireless Sensor Networks for Irrigation in Crops Using Multivariate Regression Models. In Proceedings of the 2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM), Cuenca, Ecuador, 15–19 October 2018; pp. 1–6. [CrossRef]
12. Ragnoli, M.; Barile, G.; Leoni, A.; Ferri, G.; Stornelli, V. An Autonomous Low-Power LoRa-Based Flood-Monitoring System. *Low Power* **2020**, *10*, 15. [CrossRef]
13. Alippi, C. *Intelligence for Embedded Systems*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 1–283. [CrossRef]
14. Rosero-Montalvo, P.D.; Batista, V.F.L.; Rosero, E.A.; Jaramillo, E.D.; Caraguay, J.A.; Pijal-Rojas, J.; Peluffo-Ordóñez, D.H. *Intelligence in Embedded Systems: Overview and Applications*; Springer: Cham, Switzerland, 2019; pp. 874–883. [CrossRef]
15. Guo, M..; Zhou, X. Research on the water environment capacity of Chanba River downstream. In Proceedings of the 2011 International Conference on Electric Technology and Civil Engineering (ICETCE), Lushan, China, 22–24 April 2011; pp. 4411–4414. [CrossRef]
16. Patel, H.J.; Dabhi, V.K.; Prajapati, H.B. River Water Pollution Analysis using High Resolution Satellite Images : A Survey. In Proceedings of the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 15–16 March 2019; pp. 520–525. [CrossRef]
17. Shukla, A.K.; Ojha, C.S.P.; Garg, R.D. Surface water quality assessment of Ganga River Basin, India using index mapping. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 5609–5612. [CrossRef]
18. Lin, Z.; Wang, W.; Yin, H.; Jiang, S.; Jiao, G.; Yu, J. Design of Monitoring System for Rural Drinking Water Source Based on WSN. In Proceedings of the 2017 International Conference on Computer Network, Electronic and Automation (ICCNEA), Xi'an, China, 23–25 September 2017; pp. 289–293.
19. Sowmya, C.; Naidu, C.D.; Somineni, R.P.; Reddy, D.R. Implementation of Wireless Sensor Network for Real Time Overhead Tank Water Quality Monitoring. In Proceedings of the 2017 IEEE 7th International Advance Computing Conference (IACC), Hyderabad, India, 5–7 January 2017; pp. 546–551.
20. Chen, F.; Wen, F.; Jia, H. Algorithm of Data Compression Based on Multiple Principal Component Analysis over the WSN. In Proceedings of the 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), Chengdu, China, 14 October 2010; pp. 1–4.

21. Kadir, E.A.; Irie, H.; Rosa, S.L. River Water Pollution Monitoring using Multiple Sensor System of WSNs (Case: Siak River, Indonesia). In Proceedings of the 2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Bandung, Indonesia, 18–20 September 2019; pp. 75–79.
22. Zhang, Z. Data Fusion Optimization Analysis of Wireless Sensor Networks Based on Joint DS Evidence Theory and Matrix Analysis. In Proceedings of the 2019 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Hohhot, China, 24–26 October 2019; pp. 689–6894.
23. Torres, A.J.; Quezada, M.; Carrion, L.; Coronel, I.; Barragen, A. AHP analysis to minimize the effects produced by the textile industry in the rivers of Cuenca city. In Proceedings of the 2017 IEEE Mexican Humanitarian Technology Conference (MHTC), Puebla, Mexico, 29–31 March 2017; pp. 94–101. [CrossRef]
24. De Estadística y sensos, I.N. Fasículo Provincial de Imbabura. Available online: https://www.ecuadorencifras.gob.ec/institucional/home/ (accessed on 1 January 2020).
25. Encarnación, D.; Enríquez, J.; Suarez, L. *Derecho De La Naturaleza: Caso Rio Tahuando*; Technical Report; Universidad Andina Simń Bolivar Ambato, Ecuador, 2012.
26. Liu, J.; Deng, Z. Self-tuning weighted measurement fusion Wiener filter for autoregressive moving average signals with coloured noise and its convergence analysis. *IET Control. Theory Appl.* **2012**, *6*, 1899–1908. [CrossRef]
27. Rosero-Montalvo, P.D.; López-Batista, V.F.; Peluffo-Ordóñez, D.H.; Erazo-Chamorro, V.C.; Arciniega-Rocha, R.P. Multivariate Approach to Alcohol Detection in Drivers by Sensors and Artificial Vision. In *From Bioinspired Systems and Biomedical Applications to Machine Learning*; Ferrández Vicente, J.M., Álvarez-Sánchez, J.R., de la Paz López, F., Toledo Moreo, J., Adeli, H., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 234–243.
28. Antolín, D.; Medrano, N.; Calvo, B. Analysis of the operating life for battery-operated wireless sensor nodes. In Proceedings of the IECON 2013—39th Annual Conference of the IEEE Industrial Electronics Society, Vienna, Austria, 10–13 November 2013; pp. 3883–3886.



sensors

an Open Access Journal by MDPI

Indexed in: PubMed

Indexed in: Ei Compendex

CITESCORE 5.0 SCOPUS

IMPACT FACTOR 3.275

CERTIFICATE OF ACCEPTANCE

Certificate of acceptance for the manuscript (**sensors-935734**) titled:

Environment Monitoring of Rose Crops Greenhouse Based on Autonomous Vehicles with WSN and Data Analysis

Authored by:

Paul D. Rosero-Montalvo; Vanessa C. Erazo-Chamorro; Vivian F. López-Batista; María N. Moreno-García; Diego H. Peluffo-Ordóñez

has been accepted in *Sensors* (ISSN 1424-8220) on 10 October 2020

MDPI Academic Open Access Publishing since 1996

Basel, October 2020

sensors

MDPI

Article

Environment Monitoring of Rose Crops Greenhouse Based on Autonomous Vehicles with a WSN and Data Analysis

Paul D. Rosero-Montalvo [1,2,*], Vanessa C. Erazo-Chamorro [3], Vivian F. López-Batista [1], María N. Moreno-García [1], Diego H. Peluffo-Ordóñez [4,5,6,*]

[1] Department of Computer Science and Automatics, University of Salamanca, Salamanca, 37008, Spain; vivian@usal.es (V.F.L.-B.); mmg@usal.es (M.N.M.-G.)
[2] Department of Applied Sciences, Universidad Técnica del Norte, Ibarra, 100150, Ecuador;
[3] Department of Technologies, Instituto Tecnológico Superior 17 de Julio, Urcuquí, 100650, Ecuador; verazo@ist17dejulio.edu.ec
[4] School of Mathematical and Computational Sciences, Yachay Tech University, Urcuquí, 100650, Ecuador
[5] Department of Engineering Corporación Universitaria Autónoma de Nariño, Pasto, 520002, Colombia
[6] Intelligence for Embedded Systems - Research Line, SDAS Researh Group; Ibarra, 100150, Ecuador,diego.peluffo@sdas-group.com
* Correspondence: pdrosero@utn.edu.ec (P.D.R.-M.); dpeluffo@yachaytech.edu.ec (D.H.P.-O.)

Received: 01 September 2020; Accepted: 10 October 2020; Published: date

Abstract: This work presents a monitoring system for the environmental conditions of rose flower-cultivation in greenhouses. Its main objective is to improve the quality of the crops while regulating the production time. To this end, a system consisting of autonomous quadruped vehicles connected with a wireless sensor network (WSN) is developed, which supports the decision-making on type of action to be carried out in a greenhouse to maintain the appropriate environmental conditions for rose cultivation. A data analysis process was carried out, aimed at designing an in-situ intelligent system able to make proper decisions regarding the cultivation process. This process involves stages for balancing data, prototype selection, and supervised classification. The proposed system produces a significant reduction of data in the training set obtained by the WSN while reaching a high classification performance in real conditions—amounting to 90 % and 97.5%, respectively. As a remarkable outcome, it is also provided an approach to ensure correct planning and selection of routes for the autonomous vehicle through the global positioning system.

Keywords: ambient intelligence; autonomous vehicles; monitoring systems; roses crops; wireless sensor networks

1. Introduction

Rose cultivation has a great impact on the economy of Ecuador, as theses flowers are exported and cover 9% of the world's market. Rose cultivation brings approximately 500 million dollars to the national budget and covers 8000 hectares in the country [1]. With the growing demand for flower farming production, a natural environment is not always the optimum to achieve the necessary crop requirements. Extreme conditions such as direct sun exposure, hail, diseases, and pests can seriously affect the quality of the product and the volume of production [2]. For this reason, large-scale greenhouses are becoming increasingly more popular, because they can modify the environmental conditions of the interior by means of lights, ventilation, heating, among others. Thus, crop production cycles can be planned based on market needs [3,4].

Additionally, floriculture production must be carried out in an efficient and sustainable manner, causing the least negative impact on the environment. Bearing this in mind, the use of technology allows for innovating processes and decisions based on previously collected information. In this manner, the use of agricultural resources and supplies can be improved. However, the process of data acquisition requires great effort, especially when it comes to the implementation of connections and the distribution of sensors [5]. In some cases, these systems are made using cables and can be complex and expensive. Furthermore, it should be possible to modify the location of the points of measurement according to the particular needs of the crop. Due to their easy implementation and increased mobility, wireless sensor networks (WSN) are an alternative in this respect. A WSN is made up of nodes that have the ability to acquire data and send such data by means of wireless protocols [6,7]. Likewise, WSNs are low-cost, low current-consumption systems, and allow for different types of networks and more flexibility in the exchange of information. As a result, they are efficient electronic systems that can cover large growing spaces. Within the greenhouses, it is necessary to use several WSN nodes that help get reliable data to represent the state of the plant and the preventive actions that can be taken to improve production [5,8]. In addition, the implementation of autonomous-vehicles allows for the mobility of the WSN node, as they can select routes and avoid obstacles [9]. Next, the WSN node will be able to collect data from the entire crop and by means of a GPS module, store said information at its respective location. As a result, a robust data analysis methodology can be implemented that can be compiled in each WSN node in order to make decisions, learn from external stimuli, and adapt to changes [10,11].

The parameters needed for the proper development of rose plants are relative humidity, temperature, electrical conductivity, precipitation, CO_2, and light intensity [12]. In many cases, these are not taken into account in the implementation of irrigation systems. In fact, the systems are generally based on timers or criteria based on the experiences of the people in charge of the production of roses [13]. Due to this, the ground within a greenhouse does not have homogeneous conditions[14]. Consequently, rose plants may modify their functional cycle, and obtain buds in very early stages, which are vulnerable to the use of fungicides with humid and foliar acids. Therefore, in some cases, the harvest time of the plant is modified and causes delays during the seasons of greater commercialization [15].

The proposed system is made up of a WSN implemented on quadruped autonomous vehicle moving inside a greenhouse. Firstly, the WSN has a set of sensors that monitors relative humidity, CO_2, room temperature, light quantity, and soil moisture. This is done in such a manner that the entire data analysis process can be incorporated and executed within each WSN node with a low consumption of computational resources. Secondly, the quadruped vehicle is designed to avoid collisions and be able to move in the greenhouse through the global positioning system (GPS). For this, there are established points within the greenhouse considered as arrival objectives. As a result of this system, there is a 90% reduction in the training data matrix acquired during the entire rose growing cycle for the classification algorithm training, which obtained a performance of 98% under simulated conditions and 97.5% in actual operation.

The rest of the document is structured as follows: Section 2 shows related works. Section 3 presents the system's design. Section 4 shows the data analysis proposal for the implementation of machine learning algorithms. The tests and results are shown in Section 5. Finally, Section 6 highlights the most relevant conclusions and future works.

2. Related Works

Typically, roses meant for export are cultivated by means of precision agriculture, which includes the following stages: (i) data collection, (ii) processing, (iii) data analysis, and (iv) decision making. In chronological form, they are presented to the most relevant in the different years. Salleh et al. [2] presents a WSN for monitoring environmental conditions in greenhouses (2013). The nodes of the WSN have solar cells and Zigbee modules for communication. Their results focus on optimizing system resources to extend the system battery. On the other hand, in the same year, Pekosawski et al. [5] present a similar application of WSN but with the objective of analyzing gases such as CO, CO_2, and CH_4. Similarly, their application uses Zigbee modules. In the following year, a trend to perform irrigation actuators takes place as explained in Mat et al. [16]. This work describes software and hardware components for the activation of water and fertilizers in a controlled way. Then, in 2015, Liang-Ying et al. [7] used GPRS systems to send data remotely and thus avoid having a local network by sending the information to a digital repository. In 2016, Liu et al. [17] introduced the first approach to data analysis through the use of prediction models. Furthermore, in 2017, Sampaio et al. [18] presented an alternative consisting in counting the hierarchical sensor nodes for data submission priority.

In 2018, Puspitasari et al. [19] presented the first real-time applications using WSNs with a minimum bit-sending error rate. The same year, Shinde et al. [15] presented an approach focused on the structure of a WSN for the Internet of Things (IoT), by sending packets in new lightweight protocols under the TCP/IP model. Finally, in 2019, Durmus et al. [20] presented a WSN with integrated mobile robots and data collection through IoT.

Given the works cited above, it can be stated that the WSN tends to use IoT connection protocols and evaluates the mobility of systems for the homogeneous acquisition of data in the greenhouse, where the optimization of resources is becoming a much debated problem. However, most cases focus on laboratory scale solutions, which—given their dimensions—are able to be processed by a traditional WSN. Furthermore, the electronic system has no incorporated filtering and data coupling steps to minimize noise from the non-linearity of the electronic elements that make-up the sensors and carry out the conversion of a physical parameter to an electrical signal. In this respect, most of the investigations reviewed have no clean or prepared dataset that can be directly and successfully processed through decision-making systems (e.g., machine learning algorithms). In addition, they do not address the effect that the implementation of this technology has on their products. This is in fact the rationale behind the proposed system, as the developed embedded systems are intended to keep a good trade-off between the machine learning algorithms requirements and computational resources. Consequently, the proposed data analysis methodology is of great importance in demonstrating the advantages of implementing this system in real conditions.

3. Electronic Design

This section outlines the electronic design of the vehicle fitted with sensors as well as the sensor network, which is—as a whole—referred to as the system. By design, the development of the proposed system consists of three main stages: sensor network design (Section 3.1), design of the quadruped autonomous vehicle (Section 3.2), and its motion-planning alongside the WSN topology design (Section 3.3). A schematic diagram of the proposed system is shown in Figure 1.

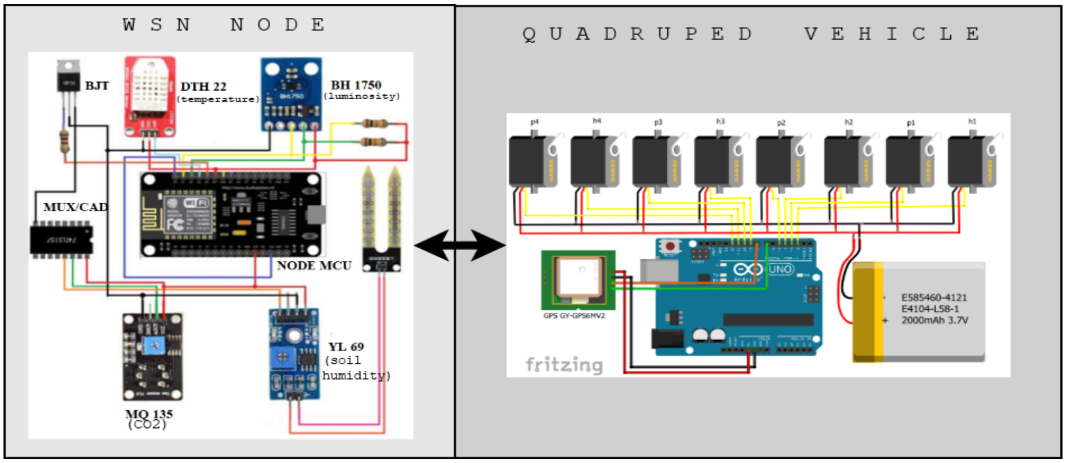


Figure 1. Schematic diagram of the proposed embedded systems for both the wireless sensor network (WSN) node (left side) and the quadruped vehicle (right).

3.1. Sensor Network Design

The environmental parameters within a greenhouse for a rose plant are temperature, luminosity, ground humidity, relative humidity, and CO_2. Each of them influences the proper growth of the rose [21]. Consequently, if the temperature is lowered below the recommended value, this can cause irregularities in the rose blossom; if it is higher, the flowers increase in number, but their quality is affected. In addition, light intensity, relative humidity, and CO_2 directly influence the photosynthetic process. Likewise, ground humidity defines the amount of water contained in unit volume of soil. If its value is not adequate, there is less chance of increasing evapotranspiration (loss of water due to heat and crop perspiration) [22,23]. The optimal values for environment variables are: room temperature 17 ^{o}C – 28 ^{o}C, light intensity 440 lx–680 lx, ground humidity 55%–65%, relative humidity 70%–80%, and CO_2 800 ppm–900 ppm [23].

For the adequate development of the WSN, the selection of the sensors must be done based on strictly-defined operating requirements. Among the main ones that were taken into account are reliability, precision, availability, ease of use, and scalability. In addition, for the selection of the WSN processor system, wireless connectivity and the protocol used were considered [14]. As a result, the sensors chosen were: `DHT 22` (relative humidity and temperature), `MQ 135` (CO_2), `YL 69` (soil humidity), `BH 1750` (luminosity). Finally, the `NodeMCu` was selected as a processor for its communication to WiFi networks. This way, data can be flexibly sent within an AD-HOC network created by the same devices, a greenhouse network, or stored in the cloud. As additional elements, an analog–digital multiplexer/converter (8 channels multiplexed at 1 output) is required in order to be able to read several sensors. This is because the `NodeMCu` only has a digital–analog converter. Additionally, a bipolar junction transistor (BJT-NPN) that works in cut and saturation is used to activate the sensors. For the power supply of the system, there is an LiPo type battery with a self-charging system that works by means of a battery manager and solar panels. Based on the datasheets of each of the electronic elements used, there is an approximate total consumption of 260 milliamps. The battery manager used is `Lio Rider` which supplies power from a solar panel until it reaches 400 milliamps. If this value is not enough, the battery can also get charged via USB-port.

3.2. Quadruped Vehicle Design

There is a wide variety of methodologies for the uniform sampling of crops. In relation to the proposed system, a grid-shaped path is defined, since it is effective with rose crops planted in a greenhouse and the vehicle can function properly [24].

The quadruped vehicle uses open hardware of the mePed version [25]. Its files of dimensions and cut-off points are free to modify in size and usability. In this case, the scale was increased 2.25 to have servo motors of greater force (8 in total). In addition, for route planning, it has a `Neo 6m` (GPS) module for its location, an `MPU-6050` sensor (accelerometer) to determine its orientation, a 3000-milliamperes LiPo battery, and an `ArduinoUno` as a processor. The armed vehicle used for testing is shown in Figure 2.

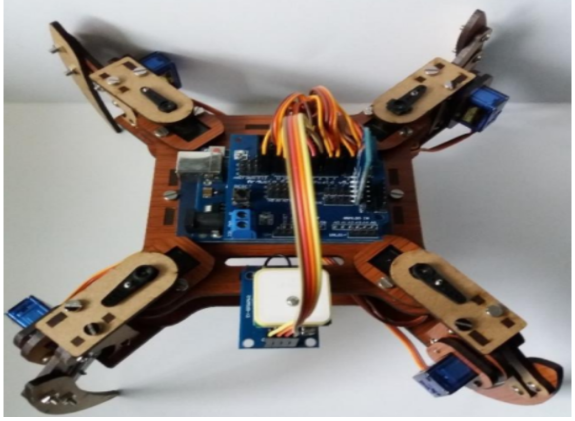


Figure 2. Quadruped vehicle in charge of collecting the greenhouse information.

For notation purposes, henceforth the term "marks" is used to refer to the vehicle's turn, while "sub-marks" accounts for sampling. Inside of the greenhouse, the vehicle operation can be divided into two stages. The first stage is the movement and location of marks and sub-marks. Each of them is marked with their coordinates where the vehicle must arrive.

To achieve this, we use the active potential field, which means that an action vector is defined to direct towards the desired mark. This criterion has two imaginary fields (attractive potential and repulsive potential). The result is a simple real-time route planning approach. The action vector is found by applying a scalar based on the potential field to the position of the vehicle and then the gradient of that function [9]. Figure 3 shows attractive potential field action vectors pointing to the goal and theirs Equations 1 and 2.

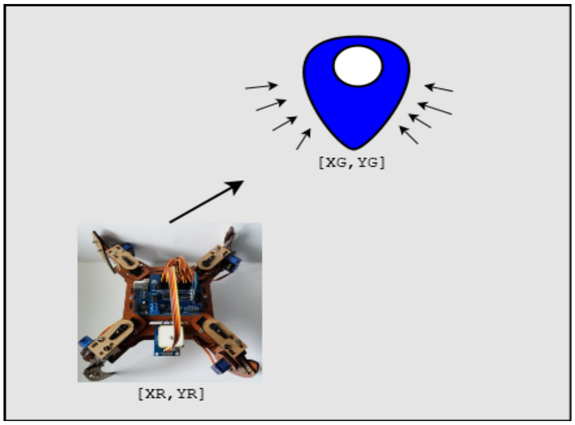


Figure 3. Potential field action vectors, white circle (goal), and blue region (potential field).

For the following statements, the following notation is considered: $[X_G, Y_G]$ as the position of the goal, r as the radius of the goal, $[X_R, Y_R]$ as the position of the vehicle, s as the size of the goal's area of influence, and α as the strength of the attractive field $(\alpha > 0)$. In this context, $[\nabla x, \nabla y]$ are the coordinates of the autonomous vehicle's movements.

The distance d from the mark to the autonomous vehicle is:

$$d = \left[(X_G - X_R)^2 + (Y_G - Y_R)^2 \right]^{\frac{1}{2}}, \tag{1}$$

while the angle θ in between the two, is given by:

$$\theta = tan^1 \left(\frac{Y_G - Y_R}{X_G - X_R} \right). \tag{2}$$

As a result, the following planning rules are feasible:

- If $d < r$, then $\nabla x = \nabla y = 0$.
- If $r \leq d \leq s + r$, then $\nabla x = \alpha(d - r)\cos(\theta)$ and $\nabla y = \alpha(d - r)\sin(\theta)$.
- If $d > s + r$, then $\nabla x = \alpha s \cos(\theta)$ and $\nabla y = \alpha s \sin(\theta)$

The second stage is soil moisture acquisition, which is carried out by the proper sensor. To do this, the vehicle reaches every sub marks, the sensor located below the chassis can be inserted with the weight of the vehicle itself. Subsequently, the system sends an activation flag to the WSN to acquire data for the rest for the sensors. This process occurs every 3 m. Figure 4 shows the route of the vehicle with the established marks and sub marks. It should be noted that the cultivation area is 50 m long and 30 m wide and there are 1-meter wide cultivation beds located 60 cm apart.

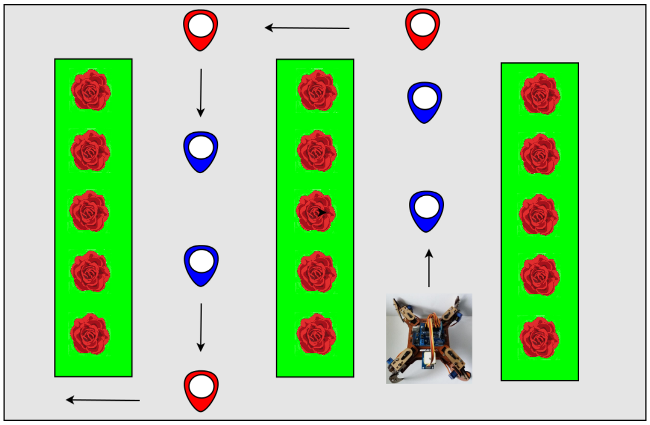


Figure 4. Vehicle movements inside to greenhouses, marks: red, sub-marks: blue.

3.3. WSN Topology

The proposed WSN has three autonomous nodes that use the sub-marks to cover the entire cultivated surface. This assignment of marks are done through the main node, which is in turn in charge of generating the routes and dividing the system based on all the previously acquired coordinates of the greenhouse dimensions. To do this, a point to multi-point topology is established, where the central node is a `Raspberry Pi 4`. A desktop computer or server is not used as a central node because the system requires a certain mobility. The process of sending data is carried out through WiFi communication, since all nodes handle this protocol. The proposed WSN topology is shown in 5.

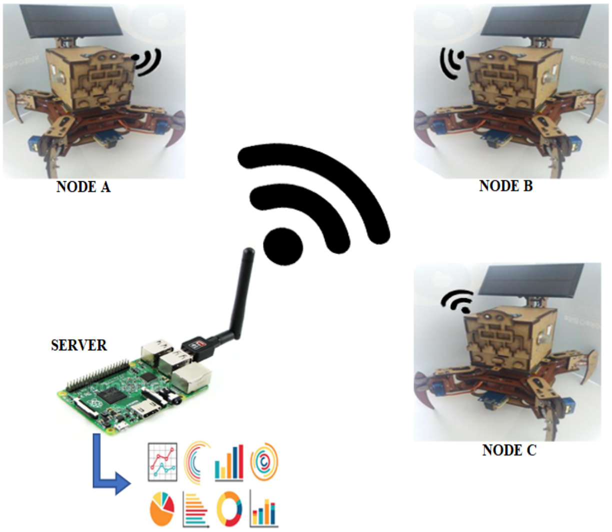


Figure 5. WSN topology with WiFi transfer data protocol and Raspberry Pi as a server.

4. Data Analysis

In this work, the data analysis is focused on reducing the instances of a high volume data matrix to achieve an improved matrix in order to be used for the training of a supervised classification algorithm and that this can be implemented in each WSN node (as seen in 4.1). In such vein, whether reaching a classification performance similar to that obtained when using the high-volume matrix: the more reduced the data matrix, the better the data representation technique. In this way, each node can make its own decision and send this information to the central node. Therefore, the aim was to find the least amount of data representing the studied phenomenon. There exists a wide range of different criteria to perform the task of selecting characteristics. In this work, data balancing (4.2) is implemented to avoid over-training regarding majority-class samples. In addition, a prototype selection technique (4.3) is used in order to eliminate data that does not provide important information to the classifier [26]. Finally, a comparison of the classical supervised classification criteria (4.4) is made to choose the appropriate algorithm that maintains a high compromise between the classification performance and the computational cost that the decision represents.

4.1. Data Acquisition

An acquisition stage consisting of coupling and filtering data is proposed with the aim of eliminating reading errors. Following from this, the DC-voltage components that can occur due to the non-linearity of the sensors are eliminated. In this sense, the average filter is preferred, which works as follows: It samples by windows of size d on a input vector $\mathbf{x} = [x_j]$ to find the average per window to yield a single point of smoothing vector $\mathbf{y} = [y_k]$. This filtering process is known as windowing or dynamic average and can be expressed as:

$$y_k = (2n+1)^{-1} \sum_{i=k-d}^{k+d} x_j, \tag{3}$$

where $\mathbf{x} = (x_1, \ldots, x_{L_x})$ is the input signal (50 samples are acquired at each established monitoring mark), $\mathbf{y} = (y_1, \ldots, y_{L_y})$ is the filtered signal, d is the window size, and L_x and L_y are respectively the input and filtered signal lengths. To account for a reduction in the computational resources usage, we experimentally define $d = 10$.

In order to establish the different environmental conditions of the crop, we created micro-environments that represent the both appropriate and harmful conditions within a greenhouse. The construction of the three micro-environments is shown in Figure 6.

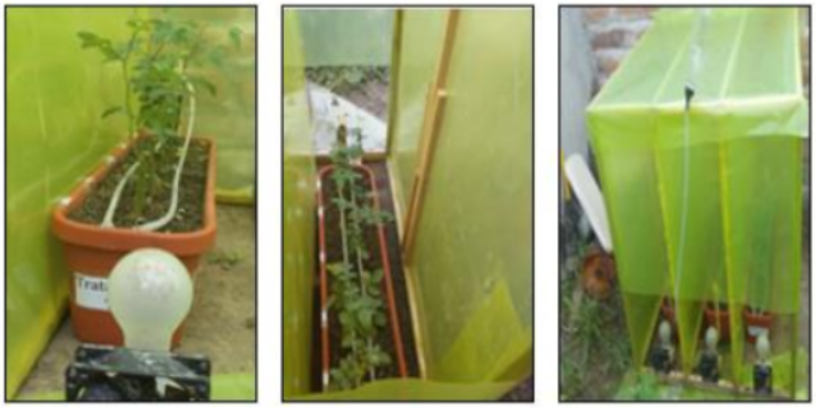


Figure 6. Rose crop micro-environments.

By using such a technique, the environment can be changed quickly without affecting a large harvest. It was possible to increase the amount of irrigation water, vary the ambient temperature, and have different amounts of CO_2 to observe their effect on roses throughout the entire growth process until the harvesting stage. At the end of this process, we obtained the classes for the database, and the subsequent learning of the system. Table 1 relates the labels of each class, their respective representation, and the modification made in their treatment to achieve different cultivation conditions. It should be considered that a greenhouse where roses are grown should have a drip irrigation system, ventilation curtains, and sunshine [23].

Table 1. Roses crop micro-environment labeling.

Label	Action	Treatment
1	Watering	Water each 8 days and curtains open only in the morning.
2	No action	Water each 4 days, curtains open in the morning and closed at night.
3	Open curtains	Water each 2 days with closed curtains.
4	Close curtains	Water each 4 days, curtains open.

After a cycle of rose crops (approximately 5 months), the matrix of data obtained is $\boldsymbol{Y} \in \mathbb{R}^{m \times n}$, where m is the number of examples and n is the number of measured environmental variables (sensors). Meanwhile, $\mathbf{L} = [\ell_i] \in \mathbb{R}^{m \times 1}$ is the label vector, with $\ell_i \in \{1, \ldots, 4\}$, $i \in \{1, \ldots, m\}$, $m = 2500$, and $n = 5$. It is important to highlight the fact that the original input data matrix cannot be stored into the WSN node because of its large size. Given the aim of improving representation and maintaining relevance, it is required that a data representation stage be carried out in advance.

4.2. Data Balancing

Under a criterion of supervised classification, an algorithm learns in relation to past instances. As a result of this, if there is a class with a greater number of data than the rest in a high volume of information, the outcome may result in an over-trained classifier which may favor the prediction of a specific label (class) in new incoming data. As a result, the classifier loses its sensitivity in complex data. The data matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ has this drawback. For the correct label assignment, the micro-environments generated for the study of rose cultivation, some of them (labels 1 and 4) greatly affected the growth of the plant. For this reason, the sampling time is shorter than that of the rest of the micro-environments (about half of the rest of the labels). In addition, despite the fact that similar sampling times were established, due to the effects of calibration times and the type of data to be processed within the signal smoothing algorithms of each sensor, their response time is varied. Therefore, according to [27], it indicates that the Kennard–Stone algorithm represents the best option in data obtained by sensors. This means that each label has the same samples in the training set.

4.3. Prototype Selection

A WSN has limited computing resources; the greater the amount of data to be processed, the longer its response time and the more affected the battery life. On the other hand, if all the data are used, there is a high possibility that many of them do not provide meaningful information to the classifier, which will reduce its decision capacity due to the model overfitting. In this sense, the prototype selection criterion (PS) is based on the concept that proper data pre-processing can reduce the size of the training matrix while maintaining the predictive ability of the algorithm. This does not affect the intrinsic knowledge initially stored.

PS algorithms are related to 3 approaches. We choose the most representative of each of them, namely: `Condensation:` Condensed Nearest Neighbor (CNN), Reduced Nearest Neighbor (RNN), and Selective Nearest Neighbor (SNN). `Edition:` Edited Nearest Neighbor (ENN), All-k Edited Nearest Neighbors (AENN), Iterative Partitioning Filter Method (IPF), and `Hybrid:` Decremental Reduction Optimization Procedures 2 (DROP 2), Decremental Reduction Optimization Procedures 3 (DROP3), and Iterative Noise Filter Method based on the Fusion of Classes (INFFC) [28].

4.4. Classification Algorithms

Classification algorithms can learn from different training criteria, such as: (i) distance-based (k-NN), (ii) density-based, (iii) model-based, and (iv) heuristic criteria. In this sense, we have used and evaluated a representative algorithm for each criterion, as follows: (i) k-NN, (ii) Bayesian classifier, (iii) support-vector-machines-based classifier (SVM), and (iv) decision tree [29].

5. Experimental Setup and Results

This section gathers the obtained experimental results, which are structured as follows: The embedded system developed in this work is presented in Section 4.4. To assess the behavior of each stage, the reduction of training data are discussed in Section 5.2. Then, the classification results are described in Section 5.3. Finally, the results of the implementation of the data analysis within the autonomous vehicle with the WSN and its operation of real conditions are shown in Section 5.4.

5.1. Developed WSN

With the WSN nodes setting (three in total), they were placed inside a housing made of chipboard material, this is due to its easy cutting, design and durability. In addition, function LEDs were implemented for each WSN node. When the system is running and data are detected, the green led shines; when the battery is running low, the red led is turned on. The solar panel has been placed at the top so that it can receive sunlight in the greenhouse. When there is no sunlight, it can switch to the normal battery. In Figure 7, the complete system is shown from the different viewpoints.

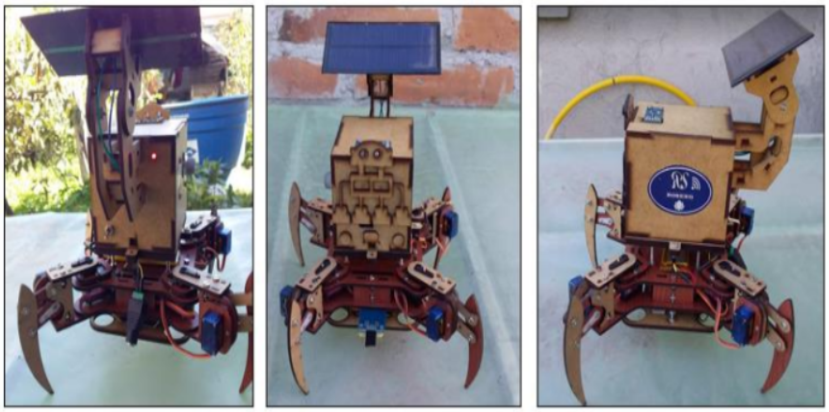


Figure 7. Implementation of the nodes onto the autonomous vehicle.

Once the WSN nodes are configured and running, the connection to the autonomous vehicle shall be implemented. These communicate through serial communication. In this way, when the autonomous vehicle reaches its mark, it sends an activation bit to the WSN to acquire the data coming from the sensors. The vehicle tilts down to bury the humidity sensor and sends that data with its GPS location to the WSN. Then, the WSN sends an end of process bit. Subsequently, the vehicle moves again until it finds another GPS mark.

With the autonomous vehicle assembled and all its parts configured, the performance tests were carried out in order to establish two aspects. The first is the battery life, where the vehicle worked without energy-saving modes in the WSN and with all the quadruped vehicle sensors running permanently. As a result, the autonomous vehicle battery lasted approximately 8 h and the WSN ones for 6 h. The second aspect refers to the sleep mode schedules implemented in the WSN. As a result of these modules, the WSN worked normally for 14 h, without the solar panel function. When it supplied extra current with the solar panel when the battery is wasted, the WSN works weeks without been turned off.

5.2. Training Matrix Reduction

By implementing the Kennard–Stone algorithm, the matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ was reduced to $\mathbf{X} \in \mathbb{R}^{p \times n}$, where $p = 1600$ (400 data per label). Subsequently, the matrix $\mathbf{X} \in \mathbb{R}^{p \times n}$ was divided into training set (80%) and test set (20%). With the new base generated, a comparison of the PS algorithms is made with the processing time metrics, removed instances and their redundant data elimination average in training set. Table 2 shows the data obtained. The CNN algorithm proved to have the best behavior in relation to the processing time and the elimination of instances. As a result, a new matrix $\mathbf{Z} \in \mathbb{R}^{s \times n}$ is defined, where $s = 128$. This process is executed by the Raspberry pi 4.

5.3. Classification Performance

A comparison of the supervised classification algorithms is carried out to determine the most appropriate ones. This is done with the classification models trained with the least-sized dataset

Table 2. Prototype selection algorithms analysis.

PS algorithm	Proc. Time (s)	Remv. Inst	% of Remv. Inst
AENN	24.02	16	1.25
BBN R	30.15	22	1.72
CNN	23.76	1152	90.0
DROP1	728.67	1120	87.5
DROP3	830.18	1120	87.5
ENN	6.14	32	2.5
RNN	27.23	1152	90.0

provided by the before-performed prototype selection stage. At first, the performance of each of them is established with the training set $\mathbf{X} \in \mathbb{R}^{p \times n}$ (data balanced for each label). Subsequently, the matrix $\mathbf{Z} \in \mathbb{R}^{s \times n}$ (Prototype selection training set) is used to observe which algorithm best represents the intrinsic knowledge of the data. It should be taken into consideration that this performance is carried out with the test of $\mathbf{X}$ (20%), which corresponds to 320 instances. In addition, SVM has some kernels that can be used; in this case, they were implemented: polynomial, sigmoid, and radial. Table 3 shows the performance of each algorithm with respect to the two data matrix.

Table 3. Performance of the considered classification algorithms.

Classification algorithms	Performance (%) on matrix $\mathbf{X}_{(1280 \times 5)}$	Performance (%) on matrix $\mathbf{Z}_{(128 \times 5)}$
k-NN	98.33	98.33
Class. Bayesian	95.27	95.27
Decision tree	95.27	93.44
SVM polynomial	95.27	95.08
SVM radial	95.08	95.08
SVM sigmoid	93.44	88.52

As can be appreciated, the k-NN algorithm reaches a higher classification performance. Therefore, different metrics to evaluate the classification performance are computed, namely: average (`acc`), error(`err`), sensitivity (`sn`), specificity (`sp`), precision (`p`), and area below the operational characteristic curve of the AUC receiver. These are good discriminators of selection of classification algorithms [29]. In table 4, the algorithm k-NN is observed with the different metrics in relation to each classification label.

Table 4. Performance values with k-NN.

Label	Acc (%)	Err (%)	Sn (%)	Sp (%)	P (%)	AUC (%)
1	100	0	100	100	100	100
2	96	3	87	97	85	92
3	90	10	57	94	57	90
4	93	7	84	95	84	90
Average	94	5	82	97	82	89

In order to obtain a graphical representation of each classification algorithm, the prediction of new instances was reduced to two dimensions through the Principal Component Analysis. This algorithm is used since it readily enables the data distribution representation in a lower dimension. Consequently, the decision edges of each label are colored. The effect of separation caused by every considered classifier can be observed in Figure 8 displayed over a lower-dimensional space for visualization purposes.

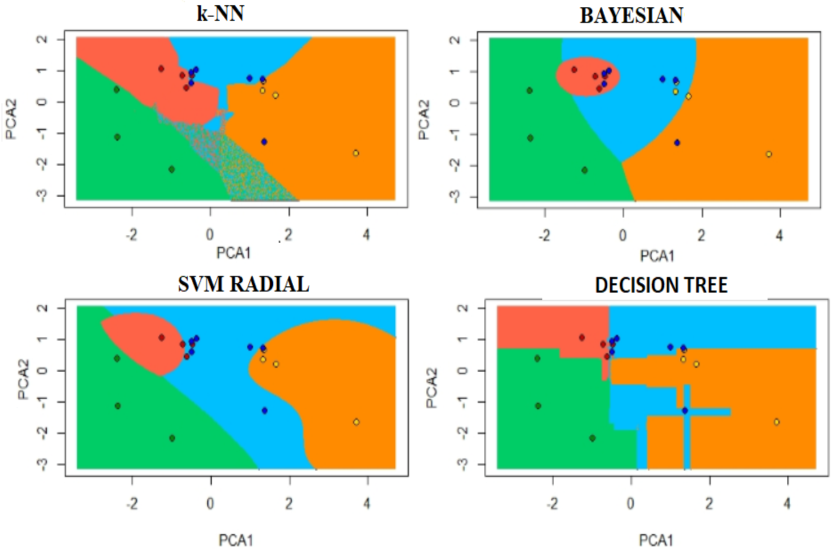


Figure 8. Decision boundaries per label of each classifier. Class 1: green color, class 2: red color, class 3: blue color and class 4: orange color. X_{axis}= Principal component 1. Y_{axis}= Principal component 2.

For the purposes of this analysis, we define a CNN as a redundant data elimination algorithm and a k-NN as a classification algorithm.

5.4. Autonomous Vehicle with WSN Implemented

First, we observe how marks and sub-marks are approached within the crop. With this, you can plan the routes of the vehicles and set their respective turn sequences. Figure 9 shows the rose crops at the beginning of their cycle and the acquisition of marks in a greenhouse.

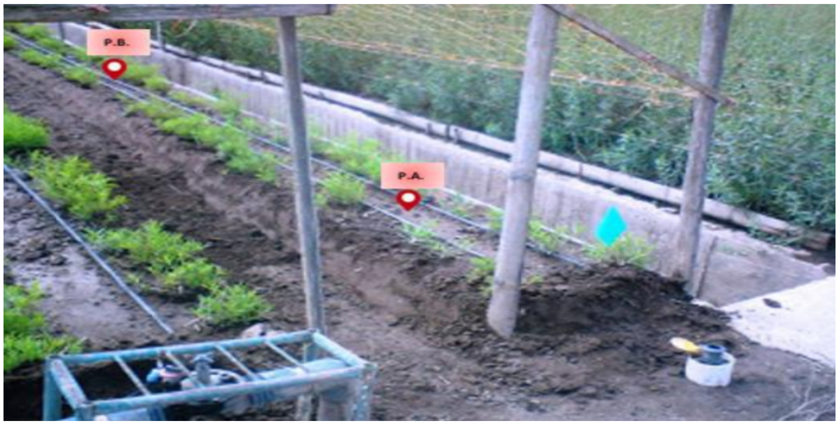


Figure 9. Sub-mark acquisition inside the greenhouse area.

The WSN node has the data stored in a matrix ***Z***; when it receives new data from the sensors, k-NN goes into operation and decides the class (set of action). Then, it receives the location of the quadruped and sends all the information in a characters vector by means a WiFi network created in the place. Subsequently, this data passes through the CNN algorithm, which verifies the valid information that can improve the training matrix. If the algorithm decides so, it is stored in the matrix **Z**; otherwise, only the sending process is performed.

Consequently, we set specific times for the compliance of routes and battery saving modes in the sensors used. As a result, the quadruped performs 4 routes a day (morning, afternoon, night, and early morning), and each route takes approximately 1 h. In addition, given its energy-saving modes, the WSN was activated only when the vehicle sent a warning about a mark that it found. For this reason, the system as a whole can operate continuously for a duration of 3 days for the quadruped and 9 days for the WSN. By having the 3 nodes working simultaneously, these vehicles can cover between 80 to 100 square meters of crops daily. The operation of the autonomous vehicle in the greenhouse is shown in Figure 10.

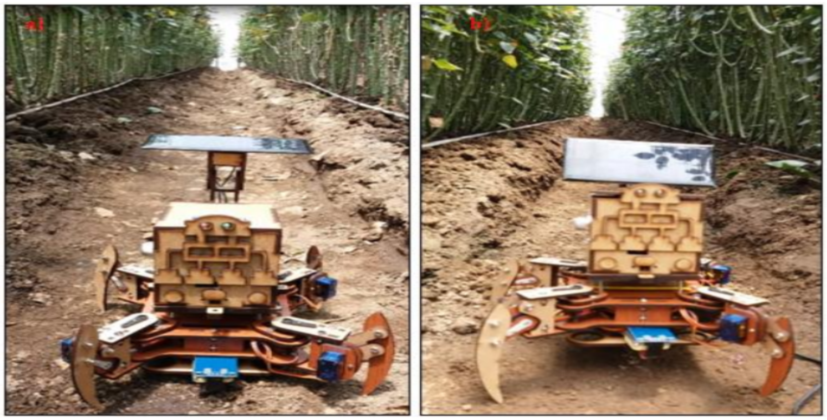


Figure 10. Autonomous system with WSN.

As a second point, it is the correct decision of the system and its assessment with respect to the chosen action by experts (people holding practical expertise and background on crops and environmental measurement equipment). These tests were performed to define the correct action of the system in the different locations of the greenhouse. Forty functioning tests were performed to assess the decision-making capacity of the system. The system had 97.5 % success in the actions taken inside the greenhouse. As a result, roses have higher stem growth and better leafiness in cultivation. In terms of return on investment, the implementation of this system had an initial margin of increasing 5 % the net profit from the crop. This is related to the lower consumption of water (15% percent), less use of pesticides (8% percent), and the result of the sale of roses (3%). It should be noted that from an economic point of view, the implementation of autonomous vehicles is well below the cost of other monitoring systems within the Ecuador–Colombia market. In addition, it was possible to verify that the greenhouses do not have homogeneous environmental conditions, there are certain sectors that due to their location in relation to the sun or distribution of the irrigation system have certain moisture deficiencies that cause less amount of CO_2 for the photosynthetic process of the plant.

Regarding the movement of the quadruped, the error in reaching each established mark has a variability of 0.35 mrts. This is due to slight variations in the collecting of GPS data related to the turn towards the other marks in the rose growing-beds. The turning angle had an error of 2 degrees in total. However, this problem is corrected by searching for the next mark inside the greenhouse.

6. Conclusions and Future Work

Related to the selection of the sensors in conjunction with the autonomous vehicle, they provided adequate operation by meeting the established marks and sub-marks inside the greenhouse. With this, the data acquisition process provided information about the crop for the implementation of the data analysis stage. This is thanks to filtering reading errors by means of data smoothing.

The WSN with autonomous quadruped vehicles fulfills the objective of providing information on the cultivation of roses by sectors within the greenhouse. Consequently, the planned scheme for data analysis was adequate, since it allowed a significant reduction of redundant data and computationally lightweight classification algorithms that can be implemented in WSN nodes with limited resources. In addition, it allows for the collection of a large amount of information that can be useful for years to come, helping farmers modify their techniques with respect to climate change.

With the autonomous vehicle, it was possible to properly arrange the growing cycles of roses. In this way, we propose a new approach to the design and construction of greenhouses, which allows for the flexibility that the crop needs (fans and floodgates in different locations, not centralized). This way, a more extensive analysis can be made with regard to change of environmental parameters in order to find optimal growth values and improve product quality.

As far as future work is concerned, we recommend exploring the use of batteries, their charge, and weight, thus ensuring better design and movement. In addition, it should be noted that the quadruped could be mobilized on irregular ground, but it had balance issues and struggled depending on the distance from the planned route. With this in mind, we suggest using other mechanisms as tank chassis.

Author Contributions: P.D.R.-M., conceptualization, methodology, software, formal analysis, investigation, writing—original draft preparation, visualization, resources; V.C.E.-C., investigation, supervision, project administration; M.N.M.-G., writing—review, investigation; V.C.E.-C., resources, electronic design, software; D.H.P.-O., validation, writing—review, methodology, supervision, project administration.

Funding: This research was funded by Fundación General de la Universidad de Salamanca en la CONVOCATORIA PARA LA REALIZACIÓN DE PRUEBAS DE CONCEPTO Y PROTECCIÓN DE RESULTADOS, PLAN TCUE 2018-2020: “Sistema Autónomo de Reconocimiento de Patrones en Invernaderos” grant number PC_TCUE18-20_021.

Acknowledgments: “Esta prueba de concepto se encuadra en el Plan TCUE 2018-2020 y ha sido seleccionada en el marco de un programa operativo cofinanciado por el Fondo Europeo de Desarrollo Regional (FEDER) y la Junta de Castilla y León.”

Conflicts of Interest: Declare conflicts of interest or state “The authors declare no conflict of interest.”

References

1. Nacional, C.F. *SECTOR AGRICULTURA, GANADERÍA, SILVICULTURA Y PESCA*. Technical report, Publisher: Quito, MINISTERIO DE AGRICULTURA Y GANADERÍ-ECUADOR, 2019.
2. Salleh, A.; Aziz, A.; Abidin, M.Z.; Misran, M.H.; Mohamad, N.R. Development of greenhouse monitoring using wireless sensor network through ZigBee technology. *Int. J. Eng. Sci.* **2013**, *2*, 6–12.
3. Toulson, R.; Wilmshurst, T. *Fast and Effective Embedded Systems Design.* Elselvier: Waltham, USA, 2017, pp. 3–18.
4. Körner, O.; Aaslyng, J.M.; Andreassen, A.U.; Holst, N. Microclimate prediction for dynamic greenhouse climate control. *HortScience* **2007**, *42*, 272–279.
5. Pekosawski, B.; Krasiński, P.; Siedlecki, M.; Napieralski, A. Autonomous wireless sensor network for greenhouse environmental conditions monitoring. In Proceedings of the 20th International Conference on Mixed Design of Integrated Circuits and Systems, Gdynia, Poland, 20–22 June 2013; pp. 503–507.
6. Jun; L.; Fu, L. Design of Greenhouse remote monitoring system based on LabVIEW. In Proceedings of the 2011 IEEE International Conference on Computer Science and Automation Engineering. Shanghai, China, 10–12 June 2011; pp. 536–539. doi:10.1109/CSAE.2011.5953277.

7. Liang, Y.; Yun, G.; Zhao, W. Greenhouse environment monitoring system design based on WSN and GPRS networks. In Proceedings of the 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER). Shenyang, China, 8–12 June 2015; pp. 795–798. doi:10.1109/CYBER.2015.7288044.
8. Luo, Q.; Qin, L.; Li, X.; Wu, G. The implementation of wireless sensor and control system in greenhouse based on ZigBee. In Proceedings of the 2016 35th Chinese Control Conference (CCC). Chengdu, China, 27–29 July 2016; pp. 8474–8478. doi:10.1109/ChiCC.2016.7554709.
9. Janos, S.; Matijevics, I. Implementation of potential field method for mobile robot navigation in greenhouse environment with WSN support. In Proceedings of the IEEE 8th International Symposium on Intelligent Systems and Informatics. Subotica, Serbia, 10–11 September 2010; pp. 319–323. doi:10.1109/SISY.2010.5647434.
10. Rosero-Montalvo, P.D.; Batista, V.F.L.; Rosero, E.A.; Jaramillo, E.D.; Caraguay, J.A.; Pijal-Rojas, J.; Peluffo-Ordóñez, D.H. Intelligence in Embedded Systems: Overview and Applications. In *Proceedings of the Future Technologies Conference*; Springer: Cham, Switzerland, 2019; pp. 874–883. doi:10.1007/978-3-030-02686-8_65.
11. Alippi, C. *Intelligence for Embedded Systems*; Springer: Berlin, Germany, 2014; pp. 1–283. doi:10.1007/978-3-319-05278-6.
12. Zheng, Z.; Wang, Y. Research on the relationship among the growth period environmental factors of tomato under the condition of mulched drip irrigation in greenhouse. In Proceedings of the 2016 Fifth International Conference on Agro-Geoinformatics (Agro-Geoinformatics). Tianjin, China, 18–20 July 2016; pp. 1–5. doi:10.1109/Agro-Geoinformatics.2016.7577648.
13. Ahonen, T.; Virrankoski, R.; Elmusrati, M. Greenhouse Monitoring with Wireless Sensor Network. In Proceedings of the 2008 IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications. Beijing, China, 12–15 October 2008; pp. 403–408. doi:10.1109/MESA.2008.4735744.
14. Rosero-Montalvo, P.D.; Pijal-Rojas, J.; Vasquez-Ayala, C.; Maya, E.; Pupiales, C.; Suarez, L.; Benitez-Pereira, H.; Peluffo-Ordonez, D. Wireless Sensor Networks for Irrigation in Crops Using Multivariate Regression Models. In Proceedings of the 2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM). Cuenca, Ecuador, 15–19 October 2018; pp. 1–6. doi:10.1109/ETCM.2018.8580322.
15. Shinde, D.; Siddiqui, N. IOT Based Environment change Monitoring & Controlling in Greenhouse using WSN. In Proceedings of the 2018 International Conference on Information , Communication, Engineering and Technology (ICICET). Pune, India, 29–31 Auguest 2018; pp. 1–5. doi:10.1109/ICICET.2018.8533808.
16. Mat, I.; Kassim, M.R.M.; Harun, A.N. Precision irrigation performance measurement using wireless sensor network. In Proceedings of the 2014 Sixth International Conference on Ubiquitous and Future Networks (ICUFN), Shanghai, China, 8–11 July 2014; pp. 154–157. doi:10.1109/ICUFN.2014.6876771.
17. Liu, Q.; Jin, D.; Shen, J.; Fu, Z.; Linge, N. A WSN-based prediction model of microclimate in a greenhouse using extreme learning approaches. In Proceedings of the 2016 18th International Conference on Advanced Communication Technology (ICACT). Pyeongchang, South Korea, 31 Janurary–3 Feburary 2016; pp. 1–2. doi:10.1109/ICACT.2016.7423608.
18. Sampaio, H.; Motoyama, S. Implementation of a greenhouse monitoring system using hierarchical wireless sensor network. In Proceedings of the 2017 IEEE 9th Latin-American Conference on Communications (LATINCOM), Guatemala City, Guatemala, 8–10 November 2017; pp. 1–6. doi:10.1109/LATINCOM.2017.8240156.
19. Puspitasari, W.; Perdana R, H.Y. Real-Time Monitoring and Automated Control of Greenhouse Using Wireless Sensor Network: Design and Implementation. In Proceedings of the 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, Indonesia, 21–22 November 2018; pp. 362–366. doi:10.1109/ISRITI.2018.8864377.
20. Durmuş, H.; Güneş, E.O. Integration of the Mobile Robot and Internet of Things to Collect Data from the Agricultural Fields. In Proceedings of the 2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics), Istanbul, Turkey, Turkey,16–19 July 2019; pp. 1–5. doi:10.1109/Agro-Geoinformatics.2019.8820578.

21. Codeluppi, G.; Cilfone, A.; Davoli, L.; Ferrari, G. LoRaFarM: A LoRaWAN-Based Smart Farming Modular IoT Architecture. *Sensors* **2020**, *20*, 2028. doi:10.3390/s20072028.
22. Erazo, M.; Rivas, D.; Pérez, M.; Galarza, O.; Bautista, V.; Huerta, M.; Rojo, J.L. Design and implementation of a wireless sensor network for rose greenhouses monitoring. In Proceedings of the 2015 6th International Conference on Automation, Robotics and Applications (ICARA), Queenstown, New Zealand, 17–19 Feburary 2015; pp. 256–261. doi:10.1109/ICARA.2015.7081156.
23. FAO. *CTA help agriculture tap into the power of digital data*; FAO: Rome, Italy, 2017. http://www.fao.org/europe/news/detail-news/en/c/1177088/
24. Domingo, A. *Cómo realizar un muestreo de suelo*; Instituto Nacional de Tecnología Agropecuaria: Ciudad de Buenos Aires, Argentina, 2015. https://inta.gob.ar/documentos/muestreo-de-suelos
25. Technologies, A.S. mePed v2 | mePed.io, 2018. http://meped.io/mepedv2
26. Hemming, S.; de Zwart, F.; Elings, A.; Righini, I.; Petropoulou, A. Remote Control of Greenhouse Vegetable Production with Artificial Intelligence—Greenhouse Climate, Irrigation, and Crop Production. *Sensors* **2019**, *19*, 1807. doi:10.3390/s19081807.
27. Rosero-Montalvo, P.D.; Peluffo-Ordóñez, D.H.; López Batista, V.F.; Serrano, J.; Rosero, E.A. Intelligent System for Identification of Wheelchair User's Posture Using Machine Learning Techniques. *IEEE Sens. J.* **2019**, *19*, 1936–1942. doi:10.1109/JSEN.2018.2885323.
28. Rosero-Montalvo, P.D.; Umaquinga-Criollo, A.C.; Flores, S.; Suarez, L.; Pijal, J.; Ponce-Guevara, K.L.; Nejer, D.; Guzman, A.; Lugo, D.; Moncayo, K. Neighborhood Criterion Analysis for Prototype Selection Applied in WSN Data. In Proceedings of the 2017 International Conference on Information Systems and Computer Science (INCISCOS). Quito, Ecuador, 23–25 November 2017; pp. 128–132. doi:10.1109/INCISCOS.2017.47.
29. Sen, P.C.; Hajra, M.; Ghosh, M. Supervised Classification Algorithms in Machine Learning: A Survey and Review. In *Emerging Technology in Modelling and Graphics*; Mandal, J.K.; Bhattacharya, D., Eds.; Springer Singapore: Singapore, 2020; pp. 99–111.

Acceptance Certificate

As Guest Editor of the Logic Journal of the IGPL (published by Oxford Journals), I certify that the paper:

"Air Pollution Monitoring Using WSN nodes with Machine Learning Techniques: A case study"

by Paul D. Rosero-Montalvo, Vivian F. López-Batista, Ricardo Arciniega-Rocha, and Diego H. Peluffo-Ordóñez

has been accepted to be published in this journal special issue devoted to selected papers from the 14th International Conference on Hybrid Artificial Intelligence Systems (HAIS 2020).

8th October 2020

Sincerely,

Emilio S. Corchado

Prof. Emilio Corchado
Guest Editor for Special Issue HAIS 2019

Logic Journal of the IGPL
Online ISSN 1368-9894 / Print ISSN 1367-0751

Air Pollution Monitoring Using WSN nodes with Machine Learning Techniques: A case study

Paul D. Rosero-Montalvo[a,b,*], Vivian F. López-Batista[a], Ricardo Arciniega-Rocha[c], Diego H. Peluffo-Ordóñez[d,e,f]

[a] *Departamento de Informática, Universidad de Salamanca, Spain*
[b] *Facultad de Ingeniería en Ciencas Aplicadas, Universidad Técnica del Norte, Ecuador*
[c] *Instituto Tecnológico Superior 17 de Julio, Yachay-Ecuador*
[d] *Yachay Tech, Ecuador*
[e] *Corporación Universitaria Autónoma de Nariño - Pasto, Colombia*
[f] *Indigo Research - Bogotá, Colombia*

Abstract

Air pollution is a current concern of people and government entities. Its monitoring and subsequent analysis in cities is a constant challenge due to variability polluting factors. For this reason, the present work shows the development of a wireless sensors network that, through machine learning techniques, can be classified into three different types of environments: high pollution levels, medium pollution, and no noticeable contamination into the Ibarra-City. To achieve this goal, signal smoothing stages, prototype selection, feature analysis, and a comparison of classification algorithms are performed. As relevant results, there is a classification performance of 95% with a significant noisy data reduction.

Keywords: wsn, air pollution, data analysis

Declarations of interest: none

1. Introduction

The different microclimates of the planet are strongly connected. This is due to different factors such as sea currents, weather, moon movement, among oth-

[*] Corresponding author

Email addresses: pdrosero@utn.edu.ec (Paul D. Rosero-Montalvo), vivian@usal.es (Vivian F. López-Batista), rarciniega@ist17dejulio.edu.ec (Ricardo Arciniega-Rocha), dpeluffo@yachaytech.edu.ec (Diego H. Peluffo-Ordóñez)

ers. These variables influence temperature, humidity, atmospheric pressure, and precipitation on different continents. In this sense, it becomes a very complex system and any alteration can cause a serious impact on the planet. In recent years, one of the biggest concerns worldwide is rising the planet's temperature. It produces climatic variations that, on the one hand, can generate excessive heatwaves that erode the ground resulting in animal and plant death.. On the other hand, aggressive rains generate floods, river overflows, among others. [1]. This is mainly due to the uncontrolled industries growth that causes the extermination of forests and generates toxic air and water pollution [2]. These industrialization effects, together with urbanization and individual mobility of people, have become a great health risk [3] [2]. Consequently, the World Health Organization (WHO) estimates that one in eight premature deaths is due to the air pollution effects [4]. With this, it can be deduced that about 3 million people die from air pollution [5]. The most polluting and detectable gases, on the one hand, is nitrogen oxide (NOx), which is a generic term to refers for a group of highly reactive gases such as nitric oxide (NO) and nitrogen dioxide (NO2) that nitrogen and oxygen content in various proportions [6].The main sources of NOx are diesel buses, power generators plants and other industrial, commercial and domestic sources that burn oil fuels. [1].In the atmosphere, nitrogen oxides can contribute to the formation of photochemical ozone (smog or polluting fog) and have health consequences. If exist prolonged or continuous exposure, the nervous system and the cardiovascular system can be affected, causing neurological and cardiac disorders. On the other hand, carbon monoxide (CO) [7], its main source is the transportation sector due to the incomplete combustion of gas, petroleum, gasoline and coal. Machines that burn fossil fuels, such as stoves or heaters. This type of contamination can lead to health conditions that can include: mental confusion, dizziness, headache, weakness and consciousness loss. It also contributes to global warming and can lead to acid rain. Both gases also act as precursors to ozone formation that potentially aggravate climatic conditions. [8] [9]

Government entities of each country have made actions to counteract climate change. One of the most important strategies is the implementation of environmental monitoring nodes located in different rural and urban sectors. With this, it is possible to have large volumes of data information to analyze them and propose strategies in the reduction of air pollutants. [10]. For this reason, the Internet of Things (IoT) is a fundamental pillar for the deployment of electronic devices that can collect data. However, this technology implementation process must sometimes be installed in sectors that are difficult to access or where the cost of wired data transfer solutions is very expensive. For this propose, the Wireless Sensor Networks (WSN), allows fulfilling the aim due to its flexibility of use, low power consumption and implementation of wireless communication protocols for its connection to a data storage server. This whole process is made from the use of sensors, which are responsible for acquiring data of the phenomenon to be studied and converting it by means of a transducer to an electrical signal that can be processed [11, 12]. However, the amount of acquired data can be received with a lot of noise. This is due to many reasons, such as the non-linearity of the electronic elements, the wear of the electronic device, among others. For these reasons, the signal must go through a cleaning and selection process to have reliable data on the phenomenon studied. In addition, due to the large area that cities cover, the vehicle density, location of companies, and proximity to natural ecosystems. The pollution index is varied within the city. For this reason, most applications that display air quality provide only an approximation of what is actually happening. To get the aim, the implementation of a WSN network makes it possible to cover large areas of land and have information by sectors that provide real information on what is happening in relation to the air pollution[13]. With this, one of the main characteristics of an integrated system must be adaptability. That is, they can emulate some processing skills that the human brain performs. The same thing implies in some way, the ability to make decisions, learn from external stimuli, adapt to changes or the possibility of executing intelligent mathematical algorithms. Implicitly, it is based on a computational paradigm that receives or processes data to accomplish an assigned task [14].

Among the pollutants mostly analyzed are nitrogen dioxide (NO2) Sulfur dioxide (SO2), tropospheric ozone (O3); in the second range are solid particles (PM10), CO, Hydrogen (H2) in consideration of the other pollutants analyzed [15, 16]. Studies such as [17], [12], [18], [19], [20] have developed data collection and monitoring systems using WSN. In addition, [21] presents a single node IoT solution, [16] shows a solution to a web platform and [20] performs data analysis with a database already acquired. With these systems development, the most commonly used machine learning algorithms are Naive Bayes, decision tree, Neural Networks, and Decision bearing machine (SVM). However, there are open problems, such as the filtering of data from the sensors, a solution that covers the entire city and provides information differentiated by sectors on existing pollution. Besides, the evaluation in the selection of the classification algorithm that best suits the phenomenon studied.

With the aforementioned, this system shows the development and installation of 13 WSN nodes strategically located in different sectors of the city of Ibarra-Ecuador. The same ones that connect to WiFi networks or 4G cellular network according to the availability of the place. With this, data acquisition is realized by means of sensors to later carry out a stage of pre-processing of the information, where criteria of signal smoothing and feature selection are used. Subsequently, a performance analysis is carried out for the selection of the classification algorithm. For this stage, 3 types of cases are defined, (i) high levels of pollution, (ii) normally the presence of gases and (iii) absence of emissions. Finally, it presents the information obtained on an IoT server to view the state of the city in real-time in relation to the location of the sensor. As relevant results, the performance of the classifier with the k-NN algorithm is 95 % under real conditions.

The rest of the document is structured as follows. The section 2 shows the electronic design and location selection strategies for each WSN node.The section 3, presents the proposed data analysis. The results are presented in the section 4.Finally, the conclusions and future works are shown in the section 5.

2. Wireless Sensor Network Design

This chapter shows the design of the WSN electronic system (2.1), and after the design is defined the locations of the WSN nodes and the method of acquisition and data filtering techniques (2.2).

2.1. Electronic Design

The WSN design focuses on monitoring the most common gases and air pollutants such as NO_x and CO. In addition, a UV sensor is used to determine the maximum rates of radiation and temperature and relative humidity sensor in order to know its relationship with air pollution. Finally, there are additional elements such as the warning LEDs and the activation button. In some cases, if a direct Internet connection is not possible, there is a GPS module and a micro-SD input to store the data locally. All these data obtained by the sensors are processed by a microcontroller with a wireless network card.

With the established requirements of functionality and accuracy of the system. The following materials are defined. The `NodeMCU` is defined as microcontroller with the sensors `MQ-7` (carbon monoxide), `MQ-135` (carbon dioxide), `ML8511` (UV rays) and `DTH11` (temperature and humidity). With this, one of the main characteristics of an integrated system must be adaptability. That is, they can emulate some processing skills that the human brain performs. The connection diagram of the WSN node is shown in Fig. 1

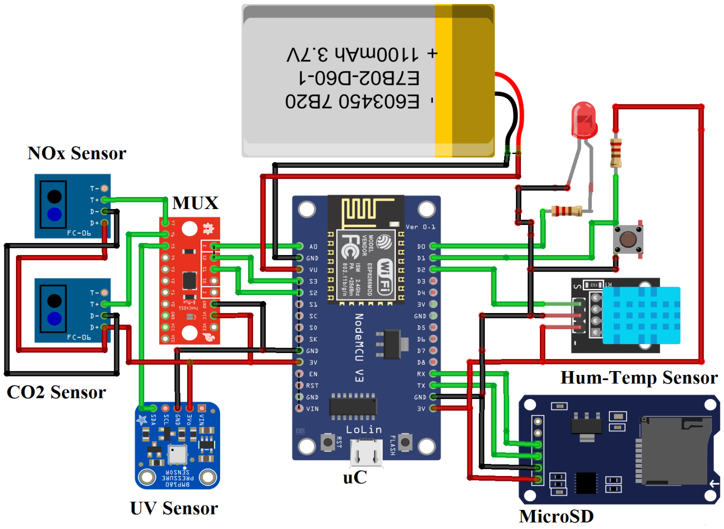


Figure 1: Electronic system developed

Once the electronic system is designed, proceed whit the calibration of each of the sensors, especially those that acquire data from ambient gases. In this sense, the `MQ-7` and `MQ-135` sensors have an internal resistance that, in relation to its sensitivity curve as a function of the power of the type $y = ax^b$, must be configured for the detection of CO and NO$_x$ respectively. To do this, the equation 1 [22].

$$ppm = \left[\frac{\frac{R_s}{R_o}}{a}\right]^{\frac{1}{b}} \tag{1}$$

Where the values of $a = 20.6690525600$, $b = -0.656039042$ for CO detection $a = 5.5973021420$, $b = -0.365425824$ for NO$_x$ detection with resistance of $R_s = 10k\omega$, $R_o = 100ppm$. For their part, the sensors `DTH11` y `ML8511` have their own libraries for adequate data conversion to the corresponding units. Consequently, sensors do not need to be calibrated.

In this sense, following reliability criteria for each sensor, some recommended performance measures are considered, such as: (i) Precision: the ability to provide the same reading by repeatedly performing the same experiment (standard deviation), (ii) Reproducibility: capacity to reproduces the same results by modifying the initial conditions of the experiment, and (iii) Stability: the ability to produce the same output value in a long time. The general results obtained are compiled in Table 1, which corresponds to 10 tests in controlled environments to evaluate the data stability. As can be seen, the data collected from the sensors show an average error of 8 % in contrast to those obtained with different available environmental monitoring applications such as Plume,Air Quality,GAIA air station, among others. Such an error is acceptable enough for implementation purposes.

Table 1: Sensor performance metrics.

	Sensors			
Measure	MQ-7 (CO)	MQ-135 (NO$_x$)	DTH11 (temperature)	ML8511 (UV rays)
Precision	9 ±	7 ±	8 ±	7 ±
Reproducibility	It is necessary to wait up 10 seconds for calibration to be done		Adequate	Some reading errors
Stability	4±, variable for each test	2±, variable for each test	Adequate	Adequate

Because the sensors have slight reading errors and the most reliable data possible is desired. The signal smoothing criterion is used. Since this will eliminate the components in direct voltage and outliers considered as reading noise. In this case, a comparison will be made between the most relevant algorithms: mean, mean filter, Savitzky-Golay, Kalman and Gaussian for the data from the digital sensor [23]. Finally, the sensor with the best signal-to-noise ratio (SNR) will be chosen.

2.2. WSN location and Data acquisition

The Ibarra city, is located north of the inter-Andean region of Ecuador, is a valley crossed by the Tahuando River, southeast of the Yahuarcocha Lagoon and is located at an altitude of 2215 meters above sea level. For the air pollution analysis, 3 zones are identified in the city. The first zone is the commercial part of the city, the second zone is the residential and educational one, and the third zone is located in the suburbs of the city where there are many green mountainous places. In this sense, 5 nodes are installed in zone 1 to have maximum data pollution due to the constant traffic density in the city and 4 in zones 2 and 3 with the aim of having normal pollution data and pollution-free .

Where the label assignment is defined according to air quality indices (AQI). This set of values that the AQI can take, is grouped into intervals to which a pattern or characteristic color of the air quality of a given area is associated.For this reason, the label Good (green) is defined to the obtained data in the early morning hours by the nodes of zone 3, Improvable (yellow) to the obtained data by the nodes of zone 2 when there is no vehicular density High and Bad (red) to the obtained data by the nodes located in zone 1 in hours of higher traffic density. This seeks to have a classifier system that shows the AQI assessment for each node in real-time. To do this, the data from each node is sent to an external server for its analysis stage. Where, after 2 months of data acquisition, a matrix is obtained. matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$, where m is sample numbers and n represents the attributes of phenomena studied (sensors). Meanwhile, $\mathbf{L} \in \mathbb{R}^{m \times 1}$ is a characteristic vector. In this case, $m = 12000$ y $n = 5$.

3. Data analysis

This section shows the proposed data analysis scheme. Where it is shown in the data cleansing by means of the selection of prototypes (3.1). Subsequently, the selection of characteristics that will determine the variables to be used by the classification algorithm (3.2). Finally, the classification algorithms are shown in relation to their operational criteria (3.3).

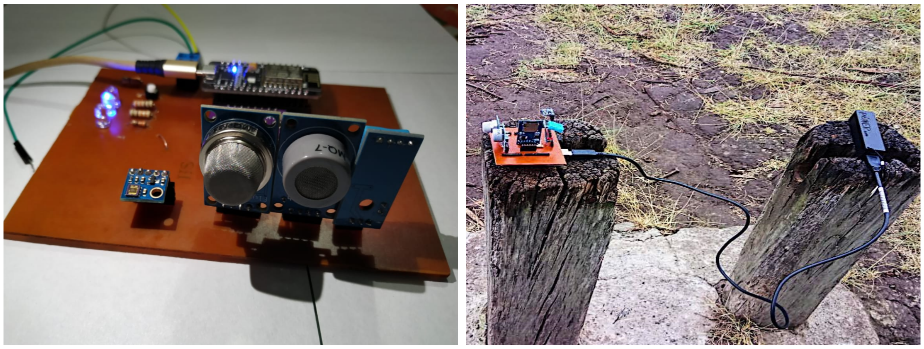
3.1. Prototype Selection

The prototype selection stage is carried out with the objective of having a training base in each WSN node. In this way, each of them can make their own decision based on their own experience. To do this, it is necessary to significantly reduce the obtained data, taking into account that the NodeMCU microcontroller only has 64k bytes as memory RAM. In this sense, [13] showed that the Condensed Nearest Neighbor (CNN) and Decremental Reduction Optimization Procedures 3 (DROP3) algorithms have a better performance in the elimination of instances and maintaining the intrinsic knowledge of the data. That is why only these algorithms will be used for this work.

3.2. Feature Selection

Typically, for a given classification task, a vast number of attributes can be candidates for characterization purposes. However, many of such attributes may be irrelevant or redundant, and, consequently, the classification algorithm may suffer from overfitting as well as some important characteristics of supervised learning may be impaired. Thus, the overall classification might not reach the expected performance. Indeed, in many applications involving large data sets, classifiers do not work properly until the unwanted features are mandatorily removed [24].

In such a vein, the feature selection task allows for reducing the number of attributes representing a data-set that best fit a model or task of machine learning. There is a wide range of criteria and algorithms that can be used for this task. However, depending on both the nature of data and data analysis goals, some feature selection criteria may result in more suitable. In this work, the feature selection task is carried out through a one-stage approach. The filtering method called `ReliefF` is used for the feature selection itself, which is a target variable driven technique (being highly recommended for multi-label classifications)[25].



(a) electronic device developed

(b) Nodes testing data

Figure 2: Nodes sensing data

3.3. Classification algorithms

There are different algorithms, it is necessary to determine the appropriate one that can be presented to the previously acquired data set. Based on refined training, the main task of the system is the identification of environmental pollution through a supervised classification. Due to this, the classification criteria have been taken in relation to the literature review of the subject under development. The same are: by distances (k-Nearest Neighbours), based on models (Decision Support Machines) and deep learning (Neural Networks).

4. Results

This section presents the overall data analysis results, as well as the systems execution tests. Results are divided into sections: WSN node (4.1), data smoothing (4.2), Prototype selection (4.3), Feature selection (4.4), classification (4.5), and implementation (4.6).

4.1. WSN node

The WSN node is developed to be implemented in the different sectors of the city of Ibarra. It has acrylic protection for moisture and rain. The two versions of the nodes (WiFi and GPS) are shown in Fig. 2.

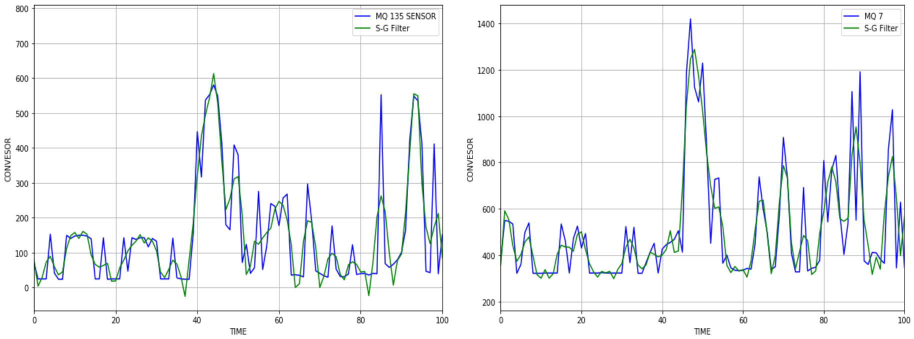
4.2. Data smoothing

The signal smoothing process uses the aforementioned algorithms and evaluates them with the signal to noise ratio (SNR). With this, it is possible to determine the algorithm better eliminates the components related to noise. It should be noted, the initial signal-to-noise ratio of the variables of each sensor is: CO = 2.12dB, NOx = 1.89, UV = 1.94, Temp = 7.12 Hum = 6.55. It should be mentioned that for the implementation of the signal smoothing algorithms, sale sizes of k value are used. Table 2 shows SNR results of each variable.

Table 2: SNR analysis

Filter	**Config.**	**SNR** (dB)				
		MQ-7	MQ-135	UV	Temp	Hum
Median	k=20	2.20	1.98	2.07	7.20	6.68
Average	k=20	2.15	2.10	2.07	7.35	6.68
Gaussian	k=20, sigma=5	2.20	2.44	2.56	8.12	7.12
Savi-Golay	k=20, pol=4	2.74	2.68	3.01	9.61	8.87

The Fig.3 shows the data smoothing results for MQ 7 and MQ135 sensor



(a) MQ 135 SENSOR

(b) MQ 7 SENSOR

Figure 3: Data smoothing for MQ7 and MQ135 sensor

4.3. Prototype Selection

In the prototype selection section, the CNN and DROP 3 algorithms were implemented. In the table 3 it can be seen that the algorithm that could eliminate more instances and has a significantly shorter time in execution is CNN.

Table 3: Prototype Selection results

PS	Remov. Inst	Remov. Inst %	Time ejec.
CNN	1050	79.16	8.25 s
DROP 3	9200	76.6	42.4 s

4.4. Feature Selection

With the selection of characteristics based on the `relief` algorithms, it was determined that the relative environmental humidity does not present any relevant information to the classifier. The UV radiation and temperature indices have a certain degree of relevance due to the hours of greatest pollution are relative to morning and afternoon hours. In the Fig. 4, the relevance of each variable is shown as a percentage.

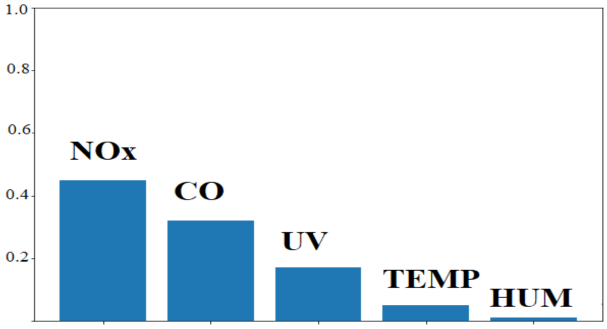


Figure 4: Feature selection analysis

4.5. Classification

In order to determine the classification algorithm, it is to choose the set of cases to induce the classifier. In this sense, it's can use the `Holdout` method, which divides the data set into two: training and testing. The test group is used to train the model and the test group to estimate the error rate. The `resampling` method becomes a generalization of the `Holdout` method, since this process is performed multiple times on different samples. With this, the error rate is based on the average of experiments performed. For this reason, the database was divided into ten different ways to train each algorithm and have an average error of each of them to get the different possible metrics from the confusion matrix.

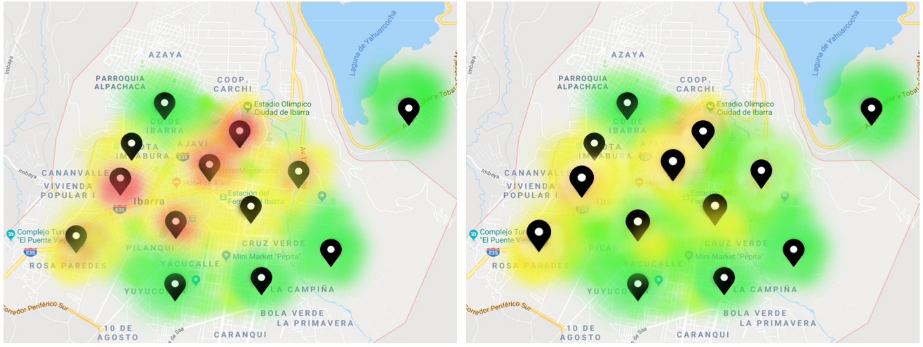
Table 4: Algorithms performance

Algorithm	Accuracy	Error Rate	Time exec
k-NN	0.95	0.5	54 ms
SVM	0.90	0.1	77 ms
Neural Network	0.96	0.4	268 ms

In this sense, two solutions can be defined in relation to the type of system operation. On the one hand, if the decision on the IoT server is wished, it is advisable to use the neural network. Since it can be compiled in a system of great computational benefits. On the other hand, if the decision is locally at each WSN node, k-NN represents better functionality and lower resource costs. In this work, the second criterion is used to search for the individual solution in each node.

4.6. Implementation

Once the entire data analysis process has been completed. The k-NN algorithm is programmed on all WSN nodes. With this, each sample they take from the air quality, they process it and make a decision. This information is sent to the IoT server that receives the data from the sensors to store it with its label and in relation to them, generates a color to show the user the level of contamination. At the moment, the server is not for public use and is investigative in nature. In the Fig. 5 shows the real air pollution measurements for each node on a normal day.

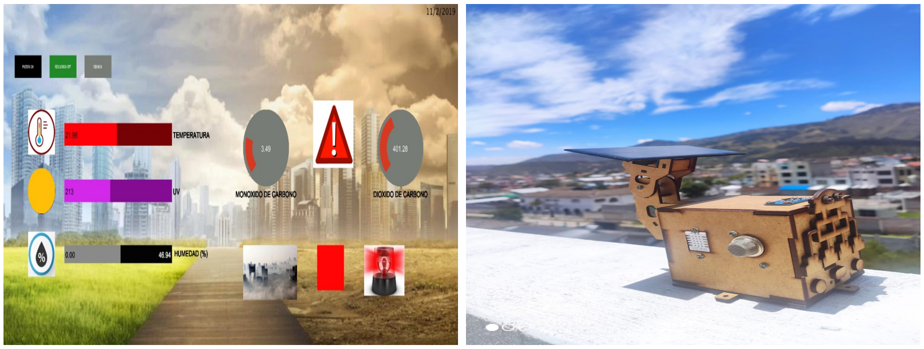


(a) Nodes sensing data at midday

(b) Nodes sensing data at night

Figure 5: Nodes sensing data

Finally, the Fig. 6 shows an individual interface created for each node to visualize data acquisition by sensors and the real conditions for one WSN node. Some nodes that are exposed to weather conditions have been reinforced with a structure of agglomerated material that internally has a silicone reinforcement to prevent water seepage.



(a) Interface developed for each WSN node

(b) Node working in zone 2 (residencial)

Figure 6: WSN nodes implementation in real conditions

5. Conclusions

The integration of the WSN nodes for the monitoring of air pollution conditions in the city of Ibarra provided important information on the sectors of greatest problem they represent. With this, planning can be made on the implementation of green areas within the city. In addition, it was possible to validate the correct functioning of the system and the way in which the machine learning algorithm adapts to the changes for decision making. In this way, the wireless protocols used (WiFi and 4G) are stable for sending data.

The proposed methodology of data analysis, starting from the data smoothing, it had the correct criteria to provide adequate information to the classifier for the training of its model and the elimination of redundant data through the selection of prototypes and feature selection.

Acknowledgment

This work is supported by the "Smart Data Analysis Systems - SDAS" group . http://sdas-group.com/

References

[1] A. K. Saha, S. Sircar, P. Chatterjee, S. Dutta, A. Mitra, A. Chatterjee, S. P. Chattopadhyay, H. N. Saha, A raspberry Pi controlled cloud based air and sound pollution monitoring system with temperature and humidity sensing, in: 2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018, Vol. 2018-January, 2018. doi:10.1109/CCWC.2018.8301660.

[2] D. Wang, E. Duan, Y. Guo, B. Sun, T. Bai, Numerical simulation of the effect of over-fire air on NOx formation in furnace, in: 2013 International Conference on Materials for Renewable Energy and Environment, IEEE, 2013, pp. 780–783. doi:10.1109/ICMREE.2013.6893790.

URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6893790

[3] K. Sujatha, N. P. G. Bhavani, R. S. Ponmagal, Impact of NOx emissions on climate and monitoring using smart sensor technology, in: 2017 International Conference on Communication and Signal Processing (ICCSP), IEEE, 2017, pp. 0853–0856. doi:10.1109/ICCSP.2017.8286488.
URL http://ieeexplore.ieee.org/document/8286488/

[4] K. Bashir Shaban, A. Kadri, E. Rezk, Urban Air Pollution Monitoring System With Forecasting Models, IEEE Sensors Journal 16 (8) (2016) 2598–2606. doi:10.1109/JSEN.2016.2514378.
URL http://ieeexplore.ieee.org/document/7370876/

[5] A. Maraj, S. Berzati, I. Efendiu, A. Shala, J. Dermaku, E. Melekoglu, Sensing platform development for air quality measurements and analysis, in: 2017 South Eastern European Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), IEEE, 2017, pp. 1–5. doi:10.23919/SEEDA-CECNSM.2017.8088233.
URL http://ieeexplore.ieee.org/document/8088233/

[6] H. Bae, Basic Principle and Practical Implementation of Near-Infrared Spectroscopy (NIRS), in: Smart Sensors and Systems, Springer International Publishing, Cham, 2015, pp. 281–302. doi:10.1007/978-3-319-14711-6_12.
URL http://link.springer.com/10.1007/978-3-319-14711-6{_}12

[7] L. Peng, F. Danni, J. Shengqian, W. Mingjie, A Movable Indoor Air Quality Monitoring System, in: 2017 2nd International Conference on Cybernetics, Robotics and Control (CRC), IEEE, 2017, pp. 126–129. doi:10.1109/CRC.2017.24.
URL http://ieeexplore.ieee.org/document/8328320/

[8] A. Quality Expert Group, Air Quality and Climate Change: A UK Perspective.

URL http://webarchive.nationalarchives.gov.uk/20130403220722/http://archive.defra.gov.uk/environment/quality/air/airquality/publications/airqual-climatechange/documents/fullreport.pdf

[9] P. D. Rosero-Montalvo, J. A. Caraguay-Procel, E. D. Jaramillo, J. M. Michilena-Calderon, A. C. Umaquinga-Criollo, M. Mediavilla-Valverde, M. A. Ruiz, L. A. Beltran, D. H. Peluffo, Air Quality Monitoring Intelligent System Using Machine Learning Techniques, in: 2018 International Conference on Information Systems and Computer Science (INCISCOS), IEEE, 2018, pp. 75–80. doi:10.1109/INCISCOS.2018.00019.
URL https://ieeexplore.ieee.org/document/8564511/

[10] A. K. Y. Law, Development of the indoor air quality index for commercial buildings in Hong Kong, Thesis, The Hong Kong Polytechnic University (2003).
URL http://ira.lib.polyu.edu.hk/handle/10397/4193

[11] Y.-L. Lin, C.-M. Kyung, H. Yasuura, Y. Liu (Eds.), Smart Sensors and Systems, Springer International Publishing, Cham, 2015. doi:10.1007/978-3-319-14711-6.
URL http://link.springer.com/10.1007/978-3-319-14711-6

[12] W. Wang, S. De, Y. Zhou, X. Huang, K. Moessner, Distributed sensor data computing in smart city applications, in: 2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), IEEE, 2017, pp. 1–5. doi:10.1109/WoWMoM.2017.7974338.
URL http://ieeexplore.ieee.org/document/7974338/

[13] P. D. Rosero-Montalvo, V. F. López-Batista, D. H. Peluffo-Ordóñez, L. L. Lorente-Leyva, X. P. Blanco-Valencia, Urban pollution environmental monitoring system using iot devices and data visualization: A case study, in: H. Pérez García, L. Sánchez González, M. Castejón Limas, H. Quintián Pardo, E. Corchado Rodríguez (Eds.), Hybrid Artificial Intelligent Systems, Springer International Publishing, Cham, 2019, pp. 686–696.

[14] P. D. Rosero-Montalvo, V. F. L. Batista, E. A. Rosero, E. D. Jaramillo, J. A. Caraguay, J. Pijal-Rojas, D. H. Peluffo-Ordóñez, Intelligence in embedded systems: Overview and applications, in: K. Arai, R. Bhatia, S. Kapoor (Eds.), Proceedings of the Future Technologies Conference (FTC) 2018, Springer International Publishing, Cham, 2019, pp. 874–883.

[15] S. M. Saad, A. M. Andrew, A. Y. M. Shakaff, A. R. M. Saad, A. M. Y. . Kamarudin, A. Zakaria, Classifying sources influencing indoor air quality (iaq) using artificial neural network (ann), Sensors 15 (5) (2015) 11665–11684. doi:10.3390/s150511665.
URL https://www.mdpi.com/1424-8220/15/5/11665

[16] D. Tudose, T. A. Pătraşcu, A. Voinescu, R. Tătăroiu, N. Ţăpuş, Mobile sensors in air pollution measurement, in: 2011 8th Workshop on Positioning, Navigation and Communication, 2011, pp. 166–170. doi:10.1109/WPNC.2011.5961035.

[17] G. B. Fioccola, R. Sommese, I. Tufano, R. Canonico, G. Ventre, Polluino: An efficient cloud-based management of IoT devices for air quality monitoring, in: 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), IEEE, 2016, pp. 1–6. doi:10.1109/RTSI.2016.7740617.
URL http://ieeexplore.ieee.org/document/7740617/

[18] N. Kafli, K. Isa, Internet of Things (IoT) for measuring and monitoring sensors data of water surface platform, in: 2017 IEEE 7th International Conference on Underwater System Technology: Theory and Applications (USYS), IEEE, 2017, pp. 1–6. doi:10.1109/USYS.2017.8309441.
URL http://ieeexplore.ieee.org/document/8309441/

[19] S. Kumar, A. Jasuja, Air quality monitoring system based on IoT using Raspberry Pi, in: 2017 International Conference on Computing, Communication and Automation (ICCCA), IEEE, 2017, pp. 1341–1346. doi:

10.1109/CCAA.2017.8230005.
URL http://ieeexplore.ieee.org/document/8230005/

[20] R. W. Gore, D. S. Deshpande, An approach for classification of health risks based on air quality levels, in: 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM), 2017, pp. 58–61, iSSN: null. doi:10.1109/ICISIM.2017.8122148.

[21] A. Kamilaris, A. Pitsillides, F. X. Prenafeta-Bold, M. I. Ali, A Web of Things based eco-system for urban computing - towards smarter cities, in: 2017 24th International Conference on Telecommunications (ICT), 2017, pp. 1–7. doi:10.1109/ICT.2017.7998277.

[22] Hwsensor, MQ7 technical data.
URL https://www.hwsensor.com/

[23] P. Kowalski, R. Smyk, Review and comparison of smoothing algorithms for one-dimensional data noise reduction, in: 2018 International Interdisciplinary PhD Workshop (IIPhDW), 2018, pp. 277–281. doi:10.1109/IIPHDW.2018.8388373.

[24] M. Peker, A. Arslan, B. Şen, F. V. Çelebi, A. But, A novel hybrid method for determining the depth of anesthesia level: Combining relieff feature selection and random forest algorithm (relieff+rf), in: 2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA), 2015, pp. 1–8. doi:10.1109/INISTA.2015.7276737.

[25] Y. Zhai, W. Song, X. Liu, L. Liu, X. Zhao, A chi-square statistics based feature selection method in text classification, in: 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 2018, pp. 160–163. doi:10.1109/ICSESS.2018.8663882.

Multivariate Approach to Alcohol Detection in Drivers by Sensors and Artificial Vision

Paul D. Rosero-Montalvo[1,2]( ), Vivian F. López-Batista[2],
Diego H. Peluffo-Ordóñez[3,4], Vanessa C. Erazo-Chamorro[5],
and Ricardo P. Arciniega-Rocha[5]

[1] Universidad Técnica del Norte, Ibarra, Ecuador
pdrosero@utn.edu.ec
[2] Universidad de Salamanca, Salamanca, Spain
[3] Universidad de Nariño, Pasto, Colombia
[4] SDAS Research Group, Yachay Tech University, Urcuquí, Ecuador
https://sdas-group.com/
[5] Instituto Tecnológico Superior 17 de Julio, Ibarra, Ecuador

Abstract. This work presents a system for detecting excess alcohol in drivers to reduce road traffic accidents. To do so, criteria such as alcohol concentration the environment, a facial temperature of the driver and width of the pupil are considered. To measure the corresponding variables, the data acquisition procedure uses sensors and artificial vision. Subsequently, data analysis is performed into stages for prototype selection and supervised classification algorithms. Accordingly, the acquired data can be stored and processed in a system with low-computational resources. As a remarkable result, the amount of training samples is significantly reduced, while an admissible classification performance is achieved - reaching then suitable settings regarding the given device's conditions.

Keywords: Alcohol detection · Drunk detection · Prototype selection · Sensors · Supervised classification

1 Introduction

The World Health Organization has reported that 40% of all road traffic accidents are caused by the drivers' drunkenness status [1]. In addition, it is the fifth main reason for deaths on the roads. As a result, 51 million people are injured or killed every year [2]. This entails a loss for expenses of approximately 500 million dollars worldwide [3]. In Ecuador, there have been registered 2100 road traffic accidents every year caused by alcohol. Unfortunately, in the last 3 years, this percentage has increased, causing a greater number of lost lives and high economic cost for society. This is because the effects of alcohol on a driver causes vision disturbances, the psychomotor function, changes in ability to react to an

J. M. Ferrández Vicente et al. (Eds.): IWINAC 2019, LNCS 11487, pp. 234–243, 2019.
https://doi.org/10.1007/978-3-030-19651-6_23

alert, behaviour and conduct [4]. Concerning psychomotor functions, the reaction time of the driver increases. This is mainly reflected when the driver needs to change the foot of the accelerator to the brake which normal time is 0.75 s, while for a driver in a drunken status, the reaction time can be 2 or more seconds [5]. As a result, the probability of suffering a road traffic accident increases considerably [6].

To counteract this high number of accidents, in Ecuador road checks are made to avoid drivers with alcohol effects transiting the roads. The main tests performed on drivers are based on the relationship between the psychological and physical faculties that a person has at the time of driving and the volume of ethanol in the body [7]. Which are based on balance tests (taking as a reference a straight line in which the driver must walk), coordination (properly locate the upper and lower extremities) and spatial perception (considering the environment in which it is located). However, the results of each test do not accurately determine the blood alcohol concentration (BAC) [8]. In addition, they present a manual approach with unlikely probabilities to detect most cases. As a result, this concentration percentage can affect the driver from least to greatest and will not be detected by the control entity. The possible alterations can be a sensation of relaxation, sedation and euphoria (0.03 to 0.05 BAC level) to vegetative state (over 0.40 BAC level).

Recent research in several biometric modalities, such as the face, the fingerprint, the iris and the recognition of the retinal area of the eye. Facial recognition is the most appropriate modality, since it is the natural way of identification among humans and is totally discreet [2,8]. However, one of the most challenging modalities in the field of artificial vision. Other studies seek to detect the parameters of alcohol in the blood through sensors [7]. The same ones that allow to share data and propose different solutions to a variety of suppliers. As a result, 4 types of knowledge have been defined for the detection of driver's driving status: (i) physiological (breathing, blood or urine), (ii) vehicle-based (road vehicle behaviour), (iii) in biological signals (cardiac and cerebral) and (iv) visual characteristics. During the last decade several studies have been carried out applying these forms of detection [5]. However, these principles can not meet the objective of an early warning that prevent the use of the vehicle by a person in drunkenness state is allowed. In addition, they can be intrusive in nature, causing discomfort to the driver, making it difficult to implement and scalability [2].

All approaches and papers presented, acquire data of the person who has the uncertainty of his ethyl state. However, many of them do not present a data analysis that allows them to make the appropriate decision and learn from the experience in different cases. In this context, the different recognition techniques have gained great acceptance in the different real-life areas [3,9]. In the field of detection of alcohol in the blood, an open touch-up is based on the application of these learning algorithms within a vehicle. In this sense, embedded systems, due to their great flexibility and portability, can be an optimal alternative. Since they seek to emulate the process performed by the human brain [10,11].

The proposed system is based on the implementation and comparison of three approaches for driver's data acquisition. In this connection, we propose to use a set of specific sensors, namely: a sensor to measure the alcohol concentration in the environment (a physiological-type one, a sensor to capture the temperature of some defined driver's face points (biological-type), and another sensor able to identify and recognize the thickness of the pupil (visual-characteristic-type). Given this, on the one hand our system seeks to eliminate the uncertainty of the concentration of alcohol in the blood. On the other hand, the system is implemented inside the car in a non-invasive way that allows to recognize the driver and monitor his/her different physical and biological signals to determine his/her suitableness to steering wheel. Consequently, it is necessary to perform a signal processing stage at a sensor level. Subsequently, a data analysis is implemented allowing for choosing the appropriate algorithm by taking into consideration the nature of the data. Finally, the system is evaluated with some performance criteria: (i) Error rate, (ii) system classification speed, and (iii) optimal usage of embedded system resources.

The rest of the document is structured as follows: Sect. 2 presents the materials and methods for the system's software and hardware. Section 3 shows the results obtained in the tests of the data analysis and the overall system operation. Finally, Sect. 4 gathers the conclusions and future work.

2 Materials and Methods

The present system is designed for the stages: (a) Design and requirements of the system, (b) The data storage scheme and (c) Data analysis.

2.1 Electronic Design

The body eliminates alcohol approximately 10 h after its intake. For this reason, the system must monitor the driver when trying to drive the vehicle. The approaches to be taken into account are presented in the scheme of Fig. 1.

As a first approach, a gas detection sensor is implemented. The same one that seeks to reveal the presence of ethanol. The selected sensor is the MQ-3, due to its sensitivity to different gases and its rapid integration into the system. Subsequently, the signal must be coupled according to the curve of the gas to be used from the digital analogue converter and configure the electric resistance Rs/Ro to convert to mg/L.

As a second approach, we seek to determine the facial thermal change of the person. since it has been proven that the temperature increases in the face of a person in the ethyl state, due to that the arteries and blood vessels increase their activity. According to [2], there are 20 different points where there is a visible variation. The same ones are: nose, eyebrows, chin and forehead. Therefore, the MLX90621 sensor is used due to its speed and temperature resolution. In addition, it has a pixel array of 16 × 4 sensitive to thermal infrared radiation.

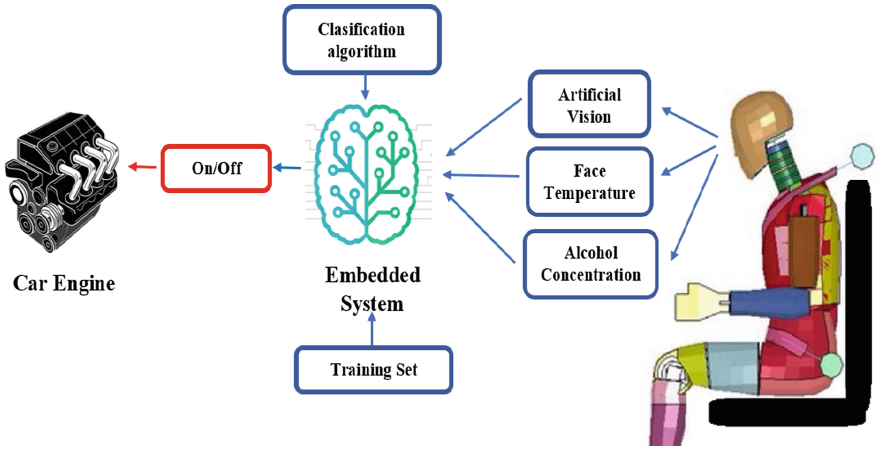


Fig. 1. Electronic system scheme

The third approach used is that of visual characteristics. Because a person who consumes alcohol the iris becomes darker, which means that its temperature compared to the sclerotic decreases. This is because the sclera is full of blood vessels that increase the temperature with alcohol consumption.

The process of acquisition of data from the pupil of the eye needs to fulfill some phases for its correct implementation. These are: (i) acquisition of images, (ii) preprocessing, (iii) segmentation, (iv) description and extraction of characteristics and (v) recognition and interpretation. Due to these reasons, there are different ways in face detection. In this system, the Viola-Jones detector has been chosen. This is because it is the one implemented by the OpenCv library that can be compiled within an embedded system.

Finally, the embedded systems that allow implementing the proposed approaches, on the one hand, the Arduino Uno for the use of the MQ-3 sensors (alcohol in the environment) and the MLX90621 for the facial temperature. On the other, the Raspberry Pi version 3 that allows to incorporate a camera for artificial vision. In addition, its computational capabilities allow the incorporation of machine learning algorithms through programming in Python.

2.2 Data Coupling Scheme

Under the criterion of a classification task, is necessary the appropriate training data set. For this reason, a stage of coupling of the sensors is performed for its correct operation [10].

As a first point, the MQ-3 sensor configures its electrical resistance for alcohol monitoring inside the vehicle. For this, data is taken in controlled tests of the environment. Subsequently, an exponential linear regression is carried out that allows to represent the concentration of the appropriate gas in representation of mg/L.

The equation to express in alcohol at a scale of 0.1 mg/L.

$$Alcohol = 0.4226 * \frac{Rs^{-1.448}}{Ro} \tag{1}$$

Once the ambient temperature and the conductor have been found, a conversion is created on a scale of colours at the measured temperature. Where the green colour shows normal value, the yellow and red colours are the temperature increase. With this, the correct position of the sensor that the information is trying to turn on the vehicle. As a result, data acquisition is stored at the minimum value and the maximum facial temperature. The Fig. 2 shows, on the one hand, the driver of the vehicle in the driving position. On the other hand, the data collection in colour scale is shown.

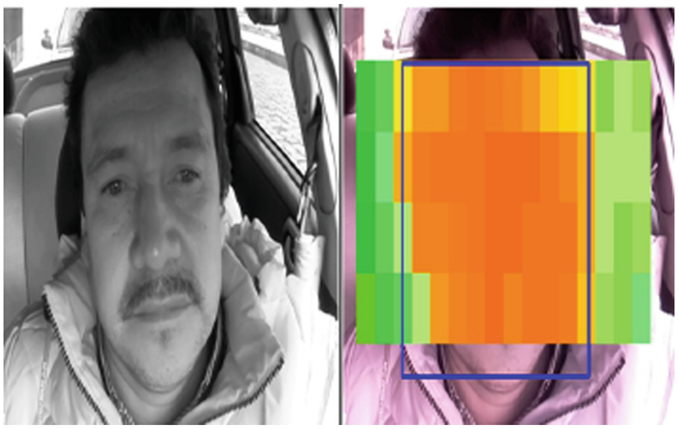


Fig. 2. Face driver temperature detection in colour scale inside at the vehicle (Color figure online)

Finally, the acquisition of images presents the following stages of preposessing: (i) Conversion of RGB to gray scale, (ii) Equalization of the histogram (improves the contrast of a magnet and normalizes the gray scale), (iii) Detection of the face, (iv) eye detection, (v) eye pupil detection and (vi) eye pupil radius. As a result, only stage (vi) is stored for data analysis.

2.3 Data Analysis

The acquired data is stored in a matrix $\boldsymbol{Y} \in \mathbb{R}^{m \times n}$, where: **m** is the number of samples and **n** represents the quantity of data acquired by the sensors and the camera. Meanwhile $\boldsymbol{L} \in \mathbb{R}^{m \times 1}$ represents the vector of the labeling of the samples. In this case $\boldsymbol{m} = 312$ and $\boldsymbol{n} = 4$. These data were obtained from 10 controlled experiments. On the one hand, the driver of the vehicle ingested different amounts of alcohol. Subsequently, an analysis of alcohol in the blood was made through the transit entity. On the other hand, the user did not ingest any kind of alcohol for data collection.

By having embedded systems of limited computational resources, the training set of classification algorithms is crucial for the response time in the detection of the assigned task. For this, the techniques of prototype selection (PS) are based on the concept that not all data provide relevant information to the classifier. There are three changes at the time of applying the prototype selection: (i) Condensation,(ii) Editions and (iii) Hybrid [9,12].

Based in a debugged training, the main task of the system is the identification of a person with alcohol in the blood through supervised classification. Due to this, the classification criteria have been taken: (i) by distances, (ii) by probabilities (iii) based on models and (iv) based on heuristics, in order to determine the appropriate one.

3 Results

This section shows the data analysis results and system implementation.

3.1 Prototype Selection

The algorithms used are in relation to the 3 approximations of PS. For this reason, the most representatives and used algorithms of each of them are chosen. **condensation:** Condensed Nearest Neighbor (CNN), Reduced Nearest Neighbor (RNN) and Selective Nearest Neighbor (SNN). **Edition:** Edited Nearest Neighbor (ENN), All-k Edited Nearest Neighbors (AENN), Iterative Partitioning Filter (IPF) and **Hybrid:** Decremental Reduction Optimization Procedures 2 (DROP 2), Decremental Reduction Optimization Procedures 3 (DROP3) and Iterative Noise Filter based on the Fusion of Classifiers (INFFC). The algorithms are executed on an I7 processor computer and 16 gigs of RAM. As a result, the result of its execution time, the reduction of instances and its percentage of elimination is shown in Table 1.

Table 1. PS data analysis

PS algorithm	Proc. time (s)	Remv. inst	% of remv. inst
CNN and RNN	6.01	290	92.94
SNN	261.15	222	71.15
DROP2	444.91	224	73.7
DROP3	399.45	292	93.58
AENN	2.66	14	4.48
ENN	0.77	10	3.2
INFFC	13.5	5	1.60
IPF	0.86	2	0.64
RNN	4.31	275	88.14

3.2 Classification Algorithms

The distance-based algorithm is considered k Nearest Neighbor (k-NN). According to the literature, the best results are obtained with k = 3 and with k = 5. A Bayesian classifier (criterion by probabilities), obtains the posterior probability of each class, C_i, using the Bayes rule, as the product of the probability *apriori* of the class by the conditional probability of attributes (E) of each class, divided by the probability of the attributes: $P(C_i|E) = P(C_i)P(E|C_i)/P(E)$. Under the model-based criteria, it uses the decision support machine (SVM) method. In this sense, the polynomial kernel function has been used: $k_P(x,y) = ((x,y) + \ gamma)$. Finally, as a heuristic criterion, the classification tree algorithm is used. Since a classifier can be defined as a function $d(x)$ defined in the classification space $\mathbf{X}$ in $\mathbf{M}$ different subsets $A_1, A_2, ..., A_M$, being $\mathbf{X}$ the union of all of them for all x belonging to A_m to the predicted class C_m.

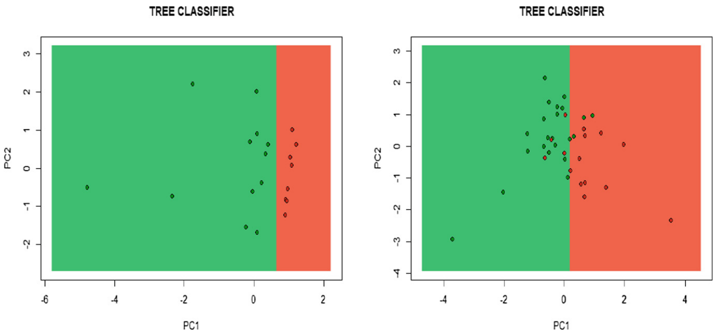
In order to define the classification criteria, the all databases within the PS section have worked. Table 2 shows the performance of each classifier with its most optimal variants and its classification average.

Table 2. PS classification performance

PS algorithm	k-NN	Clas. Bayes	Tree Dec.	SVM sigmoid
Complex data	95.15	80	97.43	96.15
CNN	80.76	45.5	79.48	94.87
SNN	79.48	55.12	97.43	93.58
DROP2	84.61	55.12	97.43	93.58
DROP3	79.48	45.5	97.43	96.15
AENN	96.15	61.15	96.15	94.87
ENN, INFFC	96.15	61.15	96.15	97.43
IPF	96.15	61.15	97.43	97.43
RNN	80.76	55.12	79.48	93.58
Average	88.58	50.06	93.58	95.12

As a result, algorithms with criteria based on heuristics and models have a high classification performance. In order to know their decision edges, the bases of greater reduction of instances were used through the reduction of dimensionality for their visualization. In Fig. 3 Decision Tree is shown.

As a result, due to the computational cost that DROP3 needs, the base to be implemented is CNN called $\boldsymbol{X} \ \ in\ \mathbb{R}^{p \times n}$, where $\mathbf{p}$ is less than $\mathbf{m}$. In this case, $\boldsymbol{p} = 22$ and $\boldsymbol{n} = 4$. In addition, the algorithm chosen by its edges is suitable for execution and its high performance is SVM. However, when performing the decision tree algorithm, it was possible to know the weighting of the variables for the classification. As a result, the concentration of ethanol with the MQ3 sensor provides more information to the classifier.

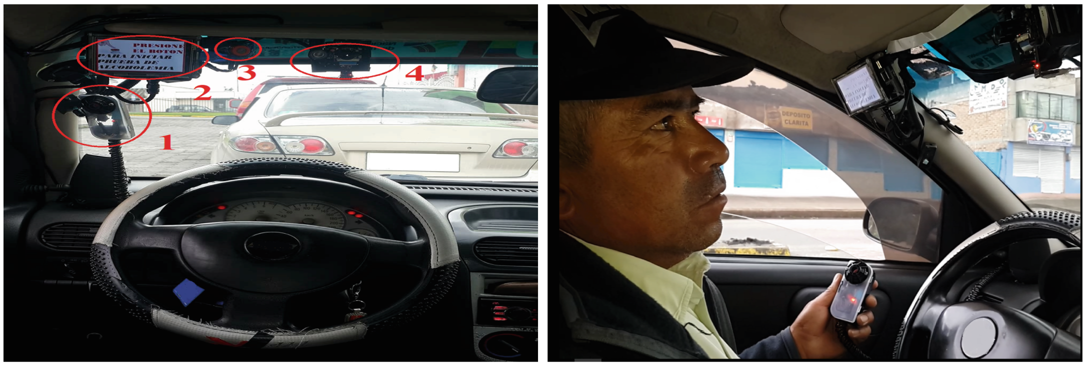


(a) DROP3 (b) CNN

Fig. 3. Decision tree algorithm

3.3 Implementation

Once the developed electronic system and the selected classification algorithms are done, the system is installed on a particular vehicle for validation. The relay module allows activating the fuel chamber of the vehicle that performs the ignition process. As a result, the system allows you to start the vehicle. In the Fig. 4 the implemented system is shown.



(a) Implemented system (b) User using the system

Fig. 4. Real test of the embedded system inside the car

Subsequently, the system is tested by the drivers of the vehicle for proper operation. In addition, an SVM algorithm is implemented in each data collection, a compiled CNN is included to improve the training matrix in each iteration.

There were 31 tests with different drivers and different drunkenness state. The system had a yield of 93.54% with a sensitivity of 100%. A specificity of 85% and a presicion of 89%. At the end of the system and with CNN's interaction in the next 20 tests, a yield of 95% was obtained. This is because the training matrix increased in value with two more instances improving the ranking algorithm.

4 Conclusions and Future Works

The criteria implemented for the acquisition data from user was correct as it allowed to satisfactorily determine their alcohol consumption. However, there were some variations in the data collection due to the variability of the environment. One of them was when the system was active with the driver's window open. This caused the surrounding gases, especially in areas of greater traffic, the sensor recognized a certain amount of these gases. The proposed methodology for the respective data analysis was adequate to represent the event in real conditions and the system can make correct decisions based on its experience. For this reason, CNN and SVM are the optimal algorithms for the acquired data set. Finally, the proposed system was not invasive to the driver and can be implemented in other types of vehicles without major inconveniences.

As future work, a phase of facial recognition of the driver is proposed to be oriented to public transport entities and mitigate more accidents on the roads in vehicles with larger numbers of passengers.

Acknowledgment. This work is supported by the "Smart Data Analysis Systems - SDAS" group (http://sdas-group.com).

References

1. Assailly, J.-P.: Young people drunk-driving: process and outcome evaluation of preventive actions. In: von Holst, H., Nygren, Å., Andersson, Å.E. (eds.) Transportation, Traffic Safety and Health - Human Behavior, pp. 297–326. Springer, Heidelberg (2000). https://doi.org/10.1007/978-3-642-57266-1_18
2. Al-Youif, S., Ali, M.A.M., Mohammed, M.N.: Alcohol detection for car locking system. In: 2018 IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE), pp. 230–233. IEEE, April 2018. https://ieeexplore.ieee.org/document/8405475/
3. Paredes-Doig, A.L., del Rosario Sun-Kou, M., Comina, G.: Alcohols detection based on Pd-doped SnO<inf>2</inf> sensors. In: 2014 IEEE 9th IberoAmerican Congress on Sensors, pp. 1–3. IEEE, October 2014. http://ieeexplore.ieee.org/document/6995514/
4. Drunk driving detection based on classification of multivariate time series. J. Saf. Res. **54**, 61.e29–64 (2015)
5. Nair, V., Charniya, N.: Drunk driving and drowsiness detection alert system. In: Pandian, D., Fernando, X., Baig, Z., Shi, F. (eds.) ISMAC 2018. LNCVB, vol. 30, pp. 1191–1207. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-00665-5_113
6. Klajner, F., Sobell, L.C., Sobell, M.B.: Prevention of drunk driving. In: Nirenberg, T.D. (ed.) Prevention of Alcohol Abuse, pp. 441–468. Springer, Boston (1984). https://doi.org/10.1007/978-1-4613-2657-1_21
7. Koukiou, G., Anastassopoulos, V.: Local difference patterns for drunk person identification. Multimed. Tools Appl. **77**(8), 9293–9305 (2018). https://doi.org/10.1007/s11042-017-4892-6

8. Wu, Y., Xia, Y., Xie, P., Ji, X.: The design of an automotive anti-drunk driving system to guarantee the uniqueness of driver. In: 2009 International Conference on Information Engineering and Computer Science, pp. 1–4. IEEE, December 2009. http://ieeexplore.ieee.org/document/5364823/
9. Rosero-Montalvo, P., et al.: Neighborhood criterion analysis for prototype selection applied in WSN data. In: 2017 International Conference on Information Systems and Computer Science (INCISCOS), pp. 128–132. IEEE, November 2017. http://ieeexplore.ieee.org/document/8328096/
10. Rosero-Montalvo, P., Peluffo-Ordonez, D.H., Batista, V.F.L., Serrano, J., Rosero, E.: Intelligent system for identification of wheelchair user's posture using machine learning techniques. IEEE Sens. J. 1 (2018). https://ieeexplore.ieee.org/document/8565996/
11. Rosero-Montalvo, P.D., et al.: Intelligence in embedded systems: overview and applications. In: Arai, K., Bhatia, R., Kapoor, S. (eds.) FTC 2018. AISC, vol. 880, pp. 874–883. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-02686-8_65
12. Rosero-Montalvo, P., et al.: Prototype reduction algorithms comparison in nearest neighbor classification for sensor data: Empirical study. In: 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM), pp. 1–5. IEEE, October 2017. http://ieeexplore.ieee.org/document/8247530/

Urban Pollution Environmental Monitoring System Using IoT Devices and Data Visualization: A Case Study

Paul D. Rosero-Montalvo[1,2,3,5(✉)], Vivian F. López-Batista[1,5],
Diego H. Peluffo-Ordóñez[2,3,5], Leandro L. Lorente-Leyva[2,5],
and X. P. Blanco-Valencia[4,5]

[1] Universidad de Salamanca, Salamanca, Spain
[2] Universidad Técnica del Norte, Ibarra, Ecuador
pdrosero@utn.edu.ec
[3] Instituto Tecnológico Superior 17 de Julio, Urcuquí, Ecuador
[4] Yachay Tech University, Urcuquí, Ecuador
[5] SDAS Research Group, Urcuquí, Ecuador

Abstract. This work presents a new approach to the Internet of Things (IoT) between sensor nodes and data analysis with visualization platform with the purpose to acquire urban pollution data. The main objective is to determine the degree of contamination in Ibarra city in real time. To do this, for one hand, thirteen IoT devices have been implemented. For another hand, a Prototype Selection and Data Balance algorithms comparison in relation to the classifier k-Nearest Neighbourhood is made. With this, the system has an adequate training set to achieve the highest classification performance. As a final result, the system presents a visualization platform that estimates the pollution condition with more than 90% accuracy.

Keywords: Intelligent system · Environmental science computing · Environmental monitoring · Data analysis

1 Introduction

The world climate conditions are very complex systems, whose it an alteration in any place has an impact on the entire planet earth. This climate change raises the global temperature and it may cause an increase in ocean waters (20 cm right now) [1]. Unfortunately, with the increase of industries and the extermination of green areas, it hopes the next century the ocean waters will increase very rapidly [2]. As a result, some cities near the coast will face flooding in 2050. In some cases, these cities will need an embankment to conserve their life. In one hand, the global temperature variation in different areas causes aggressive rains in replacement of a balanced rain [3]. Another hand, other cities of the world have been exposed to long periods of drought causing famine, animal deaths, human

H. Pérez García et al. (Eds.): HAIS 2019, LNAI 11734, pp. 686–696, 2019.
https://doi.org/10.1007/978-3-030-29859-3_58

disease, among others. Because of that, studies conducted in recent years have found the highest global temperature in comparison with other decades.

One of the principal reasons for global warming is air pollution due to the effects of industrialization, urbanization and individual mobility in vehicles, which has become a great risk to health in all countries of the world [4]. The World Health Organization (WHO) estimates that one out of every eight premature deaths is due to the effects of air pollutant particles [5]. In addition, every year there are around 3 million people who die from air pollution [6].

In order to counteract climate change, many institutions have invested in intelligent data acquisition systems to monitor environmental conditions to implement some strategies and their effect on air pollution. For do that, the Internet of Things (IoT) is fundamental, because it is in charge of developing electronic devices that collect data and exchange it with the common goal of technological advancement to improve the human conditions. In addition, to these tasks, it must have sensors to convert environmental signals to electric ones. As a result, the electronic device with sensors becomes a node of IoT cite Saha2018. However, the amount of data can be acquired with noise for many reasons like not linearity of the electronics elements, electronic device wearing, among others. For these reasons, it must go through a cleaning and selection process to choose the ideal ones that represent the phenomenon studied. As a result, measurements and monitoring of air quality are necessary for the analysis of heterogeneous environments with different emission sources like urban areas [7]. Studies such as [8–11] have developed data collection and monitoring systems. However, there are open problems, such as the appropriate decision making in remote systems, the consumption of batteries of electronic devices, adequate selection of data, among others.

The present system is installed in Ibarra-Ecuador. It can classify between 3 cases: (i) high contamination levels, (ii) Normally gases presence and (iii) No emissions. To do that, 13 sensor nodes are ubicated on the city that sends data about environmental conditions through a 4G cellphone network to an IoT server. The server allows the visualization of information on fundamental environmental parameters that must be considered when protecting the health of people. Also, the user can visualize the city situation by traffic light colors. Due to this, a prototypes selection criteria and classification algorithms are implemented in each sensor node for improving the decision task.

The rest of the document is structured as follows: Section 2 presents the design and development of the electronic system. Section 3 presents the data analysis and the implementation of the decision algorithm. Section 4 shows the results obtained in the measurement of environmental conditions. Finally, the conclusions and recommendations are presented in Sect. 5.

2 Design and Development of Electronic System

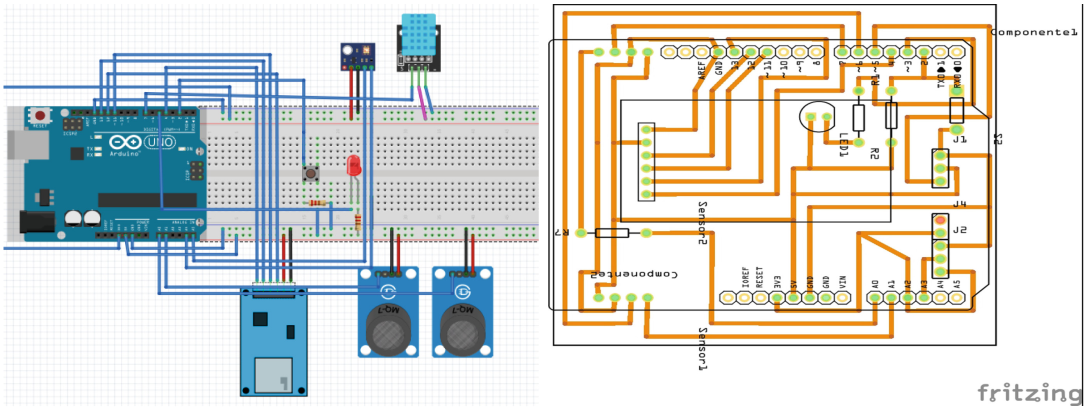
The sensor node has an Arduino UNO with PCB board with sensors MQ-7 (Carbon Monoxide), MQ-135 (Carbon Dioxide), ML8511 (UV Rays), DTH11

(Temperature and Humidity). One of the main characteristics of an embedded system must be adaptability. That is, they are able to emulate some processing skills that the human brain performs. The same that implies in some way, the ability to make decisions, to learn from external stimuli, to adapt to changes or the possibility of executing intelligent mathematical algorithms. Implicitly, it is based on a computational paradigm that receives or processes data to achieve a task assigned [12]. Under this concept, it is expected that the sensors can provide the best information as possible [13].

Related in the environmental monitoring field. The sensors that can acquire data of the most harmful and detectable gases for health caused by vehicular traffic were selected. This means, on the one hand, nitrogen oxides (NOx) which is a generic term that refers to a group of highly reactive gases such as nitric oxide (NO) and nitrogen dioxide (NO2) containing nitrogen and oxygen in various proportions [14]. The main sources of NOx are diesel buses, power plants and other industrial, commercial and domestic sources that burn fuels [1]. In the atmosphere, nitrogen oxides can contribute to the formation of photochemical ozone (smog or polluting mist) and have health consequences. If prolonged or continuous exposure occurs, the nervous system and cardiovascular system may be affected, leading to neurological and cardiac alterations. On the other hand, carbon monoxide (CO) [15] Its main source is the transportation sector due to the incomplete combustion of gas, oil, gasoline, and coal. The domestic appliances that burn fossil fuels such as stoves, stoves or heaters, among others. Your health conditions are mental confusion, vertigo, headache, nausea, weakness, and loss of consciousness. It also contributes to global warming and can cause acid rain. Both gases also act as precursors of ozone formation which potentially aggravates climatic conditions [16,17].

Finally, the system has a UV sensor to determine the maximum radiation rates and a temperature and relative humidity sensor to know its status during the day. The Fig. 1 presets the electronic connection and his PCB design. The developed system is under free hardware and software components. Its plate design is in the form of a mountable device for the Arduino (shield). In this way it can be connected and disconnected in an easy way. For the correct functioning of the sensors, a burning stage must be carried out. Finally, the system has a UV sensor to determine the maximum radiation rates and the temperature and relative humidity sensor to know its status during the day. The figure ref img1 presents the electronic connection and its PCB design.

The developed system is under free hardware and software components. Its plate design has the form of a mounting device for the Arduino (shield). In this way, it can be connected and disconnected in an easy way. For the correct functioning of the sensors, each one must be exposed to fire. In this sense, the sensor can be exposed to different environmental conditions so that the acquisition of data can be stabilized. Subsequently, the internal resistance must be configured to select the gas to be measured. This is because the harmful gas sensors have the capacity to perceive other gases. Finally, each node only acquires data every 30 min, stored it in external memory (MicroSD) as a backup, then the system sends it through the mobile network. In Figs. 1 and 2 the system developed and ready to go into operation is shown.



(a) Diagram connection (b) PCB design

Fig. 1. Electronic system and connection design

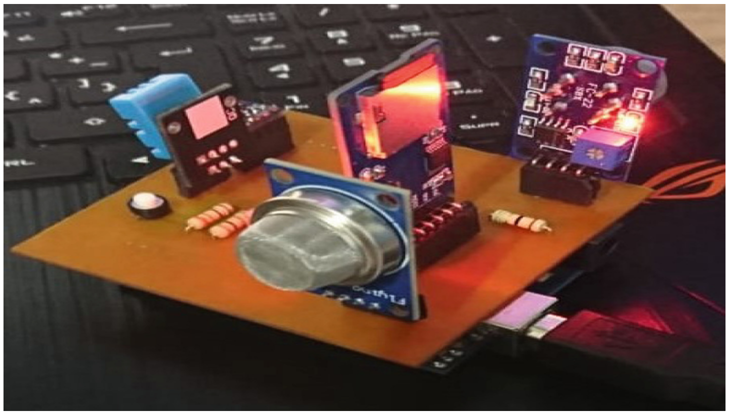


Fig. 2. Electronic system developed

3 Data Analysis

In this section presents a data analysis in two stages. The first one, a comparison with Prototype Selection and Data Balance algorithms to find new small data sets. The second one, with these training sets, is tested with k-Nearest Neighbourhood to find the best classifier performance.

3.1 Data Analysis Scheme

Sensor nodes were located in the different sectors of Ibarra city. The information collected was sent to a repository in the cloud. However, this information may be subject to different errors in reading, shipping, among others. For this reason, some machine learning criteria allow choosing the ideal data sets that represent the large volume of information acquired [18]. As a result, prototype selection algorithms focus on eliminating redundant data that does not provide information to the classifier. In this way, the decision borders of each of the categories that you want to learn will be soft and improve the classification performance [17]. Unfortunately, the selection of prototypes reducing data in an unbalanced training matrix way (it does not have the same number of instances per label). This causes the probability between classes is biased towards the majority class. Therefore, probably an error rate for new instances can reduce the classification performance. Therefore, data balancing criteria (each cluster has the same data points) allow for the training matrix to provide the same amount of information for each label [19].

Finally, under a classification criterion, the objective of the systems is to assign a new instance to a previously known set. In this sense, for the selection of environmental conditions, 3 known situations are determined. (i) A place is full of permanent contamination with high traffic flow. (ii) moderate traffic conditions with peak hours and (iii) green sectors far from the city. In Fig. 3 the proposed data analysis scheme that allows the system to make the correct decision is presented.

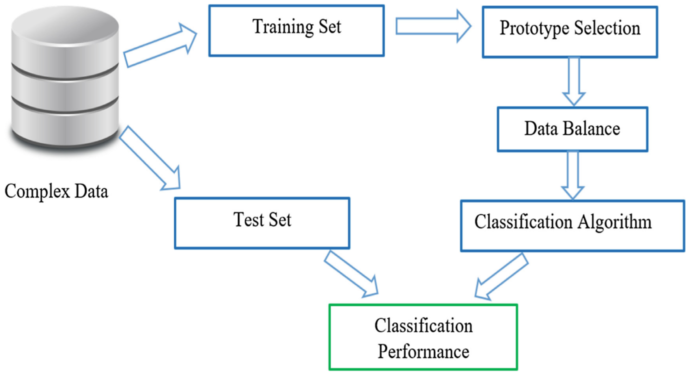


Fig. 3. Data analysis scheme

3.2 Data Analysis Criterion

Considering prototype selection criteria (PS) and data balancing (DB) are performed outside the electronic system (in a high-performance computer) different algorithms can be tested. Due to this, the most representative of each criterion is analyzed. On the one hand in PS, the algorithms to be used are All-k Edited Nearest Neighbors (AENN), Condensed Nearest Neighbor (CNN), Edited Nearest Neighbor (ENN), Reduced Nearest Neighbor (RNN), Decremental Reduction Optimization Procedures 3 (DROP3). Each one represents different ways to eliminate data (i): edition, (ii) condensations and (iii) hybrid. On the other hand, in DB the algorithms can be Kenard-Stone, Dúplex, ShenkWset and Naes [20]. All of them are implemented in different **R** software libraries. About classification algorithms, according to [20] and due to limitations of computational resources of the embedded system used (Arduino) it is advisable to use the k-Nearest Neighborhood (k-NN) classifier. For this reason, a comparison of instances removed by the PS algorithms and their execution time is performed. These data are presented in Table 1. The size of the training matrix is 1200 data points for each sensor.

Table 1. Prototype selection comparison

Algorithm	Remov. inst.	% Remov. inst.	Time ejec.
AENN	11	1.015	3.51 s
CNN	1042	96.481	2.66 s
ENN	10	0.925	3.15 s
RNN	1042	96.481	2.66 s
DROP3	1033	95.678	21.45 s

Data matrix reduced by the PS algorithms, the next step is testing DB algorithms in comparison with classification algorithm k-NN to find the best performance. As a result of each database shows in Table 2.

4 Results

CNN database has been chosen to maintain a high classification performance. In each node has been implemented a classification algorithm (k-NN) and CNN. Because of this, the own system can improve your training set. Therefore, the node sensor sends only a cleaned lecture with the decision label trough 4G network to IoT network and it going to store in local micro SD. As a final IoT scheme, thirteen nodes that acquire data were implemented. In the beginning, the first readings were systematically to achieve an optimal location of the nodes around the city. Each node was installed at least with 2 km distance of another one. As a result, it was possible to determine the ranges of affectation in relation to the decision taken by each node. With this, it was possible to show graphically the sectors with the highest concentration of gases. At Fig. 4 shows a borderline classifier label decision in two dimensions.

Table 2. k-NN performance with data balance matrix

Algorithm	Data balance	K-NN performance		
		CNN	RNN	DROP3
KENSTONE	210	0.911	0.911	0.937
DUPLEX	210	0.914	0.914	0.948
SHENKWEST	210	0.322	0.327	0.28
NAES	210	0.837	0.911	0.948

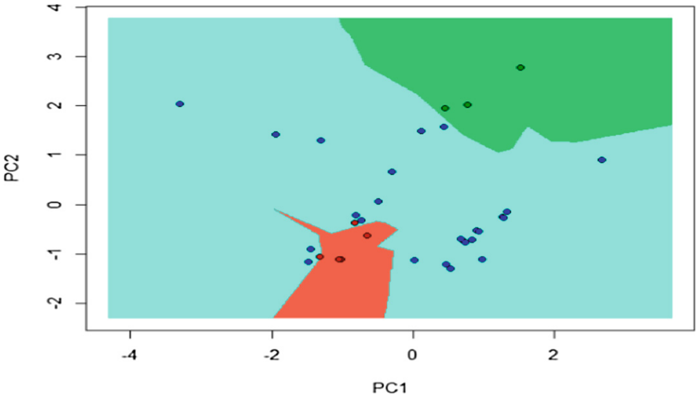
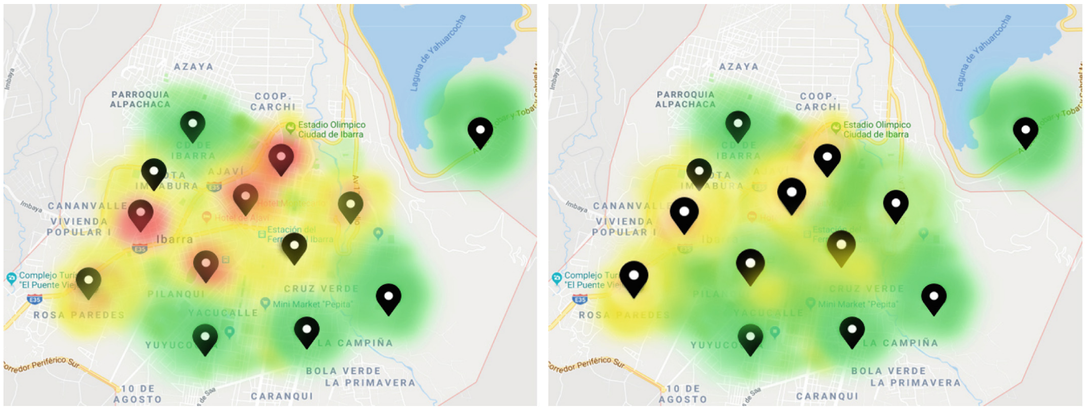


Fig. 4. Borderline classifier label decision, green: no pollution, blue: allowed range, red: caution (Color figure online)

For the correct visualization of information, IoT server was installed a Processing (visualization software). Each node sends data with their location, the visualization platform changes in relation to all of them. When the user selecting in each node, the data that is being monitored at that moment is displayed and the pollution state in colors (green: no pollution, yellow: allowed range, red: caution). For reasons of subsequent analysis, the maximum and minimum values are also stored with their time and date of all the variables to be measured. The city map and the air quality decision borders are shown in Fig. 5 Consequently, in Fig. 6 indicates the individual interface of each node. In this way, you can extract information tables with different reports in relation to the need for the study as shown in Table 3.



(a) Diagram connection

(b) Nodes sensing data at midday

Fig. 5. Nodes sensing data at night (Color figure online)

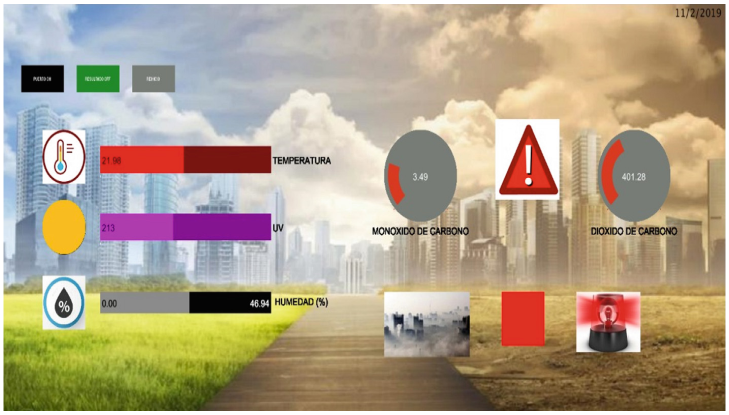


Fig. 6. Data visualization for each node in Ibarra city

Table 3. Sensor node in Bolivar street, Ibarra

Bolivar street		
	V. Max	V. Min
CO2	53.93	51.19
CO	24	22
Temperature	30°C	17°C
Humidity	68 %	40 %
UV	9.7	0

With the coordinates of the sensor nodes, it can be analyzed within a platform such as Google maps. Of this way, It can have a greater specificity of the environments your pollution status. Figure 7 shows the decision edges with greater precision in relation to traffic light color (red: high-level pollution, yellow: normal pollution and green: no pollution). Finally, Fig. 8 indicates a node with a rechargeable battery monitoring the environment in a green area.

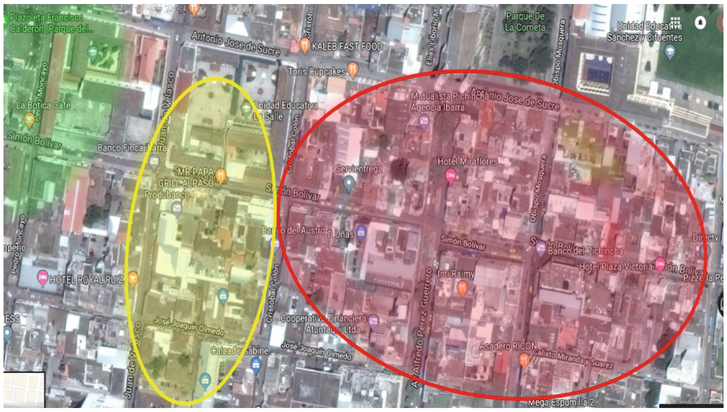


Fig. 7. Google maps with environmental analysis (Color figure online)

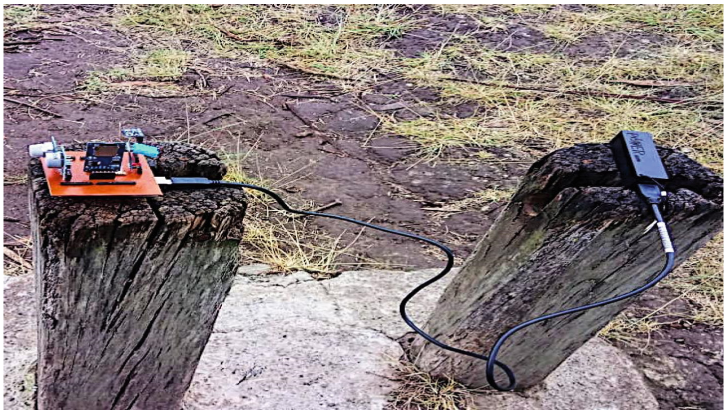


Fig. 8. Node acquiring data

As a remarkable result, the system has been tested between some specific air pollution applications. The visualization platform has more exact values because these It only shows an estimated average. In this sense, in real conditions, nodes can classify the air pollution conditions with 92% accuracy among the nodes.

5 Conclusions and Future Works

The present environmental monitoring project allowed to know the places of pollution inside the city. With this, different strategies can be planned to mitigate these problems. With respect to data analysis, the proposed proposal achieved the objective of providing the correct information to the classifier and allowed it to make an adequate decision in each node. To do this, technologies such as the IoT allow an agile collection of data and interconnect a large number of devices for the extraction of knowledge. In addition, this analysis must have an interface that allows the user to know the actions taken by nodes and be able to perform different types of analysis with the stored databases.

With regard to the analyses carried out, it was possible to deduce that the concentration of the greatest amount of harmful gases could affect up to a kilometer around where there is no emission of this type. Another point to consider is that the gases in the area of vehicular traffic by schedules gradually decrease after 25 or 30 min. Finally, the mist in cold places allows a faster dissipation by the condensation of water.

As future work, it is expected to have the IoT system actively and have nodes that can be mobile and perform a more comprehensive analysis.

Acknowledgment. This work is supported by the Smart Data Analysis Systems - SDAS group. http://sdas-group.com/.

References

1. Saha, A.K., et al.: A raspberry Pi controlled cloud based air and sound pollution monitoring system with temperature and humidity sensing. In: 2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018, vol. 2018, January 2018
2. Wang, D., Duan, E., Guo, Y., Sun, B., Bai, T.: Numerical simulation of the effect of over-fire air on NOx formation in furnace. In: 2013 International Conference on Materials for Renewable Energy and Environment, pp. 780–783. IEEE, August 2013. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6893790
3. Guariso, G., Volta, M. (eds.): Air Quality Integrated Assessment. SAST. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-33349-6
4. Sujatha, K., Bhavani, N.P.G., Ponmagal, R.S.: Impact of NOx emissions on climate and monitoring using smart sensor technology. In: 2017 International Conference on Communication and Signal Processing (ICCSP), pp. 0853–0856. IEEE, April 2017. http://ieeexplore.ieee.org/document/8286488/
5. Bashir Shaban, K., Kadri, A., Rezk, E.: Urban air pollution monitoring system with forecasting models. IEEE Sens. J. **16**(8), 2598–2606 (2016). http://ieeexplore.ieee.org/document/7370876/
6. Maraj, A., Berzati, S., Efendiu, I., Shala, A., Dermaku, J., Melekoglu, E.: Sensing platform development for air quality measurements and analysis. In: 2017 South Eastern European Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), pp. 1–5. IEEE, September 2017. http://ieeexplore.ieee.org/document/8088233/
7. Lin, Y.-L., Kyung, C.-M., Yasuura, H., Liu, Y. (eds.): Smart Sensors and Systems. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-14711-6
8. Fioccola, G.B., Sommese, R., Tufano, I., Canonico, R., Ventre, G.: Polluino: an efficient cloud-based management of IoT devices for air quality monitoring. In: 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), pp. 1–6. IEEE, September 2016. http://ieeexplore.ieee.org/document/7740617/
9. Wang, W., De, S., Zhou, Y., Huang, X., Moessner, K.: Distributed sensor data computing in smart city applications. In: 2017 IEEE 18th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1–5. IEEE, June 2017. http://ieeexplore.ieee.org/document/7974338/
10. Kafli, N., Isa, K.: Internet of Things (IoT) for measuring and monitoring sensors data of water surface platform. In: 2017 IEEE 7th International Conference on Underwater System Technology: Theory and Applications (USYS), pp. 1–6. IEEE, December 2017. http://ieeexplore.ieee.org/document/8309441/
11. Kumar, S., Jasuja, A.: Air quality monitoring system based on IoT using Raspberry Pi. In: 2017 International Conference on Computing, Communication and Automation (ICCCA), pp. 1341–1346. IEEE, May 2017. http://ieeexplore.ieee.org/document/8230005/
12. Rosero-Montalvo, P.D., et al.: Intelligence in embedded systems: overview and applications. In: Arai, K., Bhatia, R., Kapoor, S. (eds.) FTC 2018. AISC, vol. 880, pp. 874–883. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-02686-8_65
13. Chiu, S.-W., Hao, H.-C., Yang, C.-M., Yao, D.-J., Tang, K.-T.: Handheld gas sensing system. In: Lin, Y.-L., Kyung, C.-M., Yasuura, H., Liu, Y. (eds.) Smart Sensors and Systems, pp. 155–190. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-14711-6_8

14. Bae, H.: Basic principle and practical implementation of near-infrared spectroscopy (NIRS). In: Lin, Y.-L., Kyung, C.-M., Yasuura, H., Liu, Y. (eds.) Smart Sensors and Systems, pp. 281–302. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-14711-6_12
15. Peng, L., Danni, F., Shengqian, J., Mingjie, W.: A movable indoor air quality monitoring system. In: 2017 2nd International Conference on Cybernetics, Robotics and Control (CRC), pp. 126–129. IEEE, July 2017. http://ieeexplore.ieee.org/document/8328320/
16. Air Quality Expert Group: air quality and climate change: a UK perspective. http://webarchive.nationalarchives.gov.uk/20130403220722/archive.defra.gov.uk/environment/quality/air/airquality/publications/airqual-climatechange/documents/fullreport.pdf
17. Rosero-Montalvo, P.D., et al.: Air quality monitoring intelligent system using machine learning techniques. In: 2018 International Conference on Information Systems and Computer Science (INCISCOS), pp. 75–80. IEEE, November 2018. https://ieeexplore.ieee.org/document/8564511/
18. Rosero-Montalvo, P., et al.: Prototype reduction algorithms comparison in nearest neighbor classification for sensor data: empirical study. In: 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM), pp. 1–5. IEEE, October 2017. http://ieeexplore.ieee.org/document/8247530/
19. Rosero-Montalvo, P., et al.: Neighborhood criterion analysis for prototype selection applied in WSN data. In: 2017 International Conference on Information Systems and Computer Science (INCISCOS), pp. 128–132. IEEE, November 2017. http://ieeexplore.ieee.org/document/8328096/
20. Rosero-Montalvo, P.D., Peluffo-Ordóñez, D.H., López Batista, V.F., Serrano, J., Rosero, E.A.: Intelligent system for identification of wheelchair user's posture using machine learning techniques. IEEE Sens. J. **19**(5), 1936–1942 (2019)