



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Un Enfoque de Fusión de Datos de
Variables Físicas Medidas a través de
una Red de Sensores Inalámbrica**

ARTÍCULO ACADÉMICO

Que para obtener el título de
Ingeniero en Computación

P R E S E N T A

Thomas Alejandro Arias Pelayo

ASESOR DE ARTÍCULO ACADÉMICO

Dr. Guillermo Gilberto Molero Castillo



Ciudad Universitaria, Cd. Mx., 2022

Resumen

En la actualidad, los métodos de fusión de datos son una herramienta útil para integrar fuentes de datos, antes del análisis de estos, y proporcionar una vista unificada de un fenómeno o evento observado. Este documento, a manera de tesina, presenta los resultados alcanzados como parte del desarrollo del proyecto de investigación, realizado bajo la Modalidad de Titulación por Actividad de Investigación, aprobado por el Comité de Titulación de la División de Ingeniería Eléctrica de la Facultad de Ingeniería. **Objetivo.** Se propuso un método, basado en estimaciones de Monte Carlo, para la fusión de datos de variables físicas obtenidas a través de una red de sensores inalámbrica, medidas en un ambiente cerrado de ocupación humana. **Método.** Como etapa inicial, se diseñó e implementó una red de sensores inalámbrica, que fue instalada en un entorno cerrado de ocupación humana, como es el caso de una habitación de un hogar. Los datos recopilados fueron de variables físicas, como temperatura, humedad y densidad del polvo, que se almacenaron en la nube a través de ThingSpeak, la cual es una plataforma de código abierto para Internet de las Cosas. Posteriormente, se desarrolló el método de fusión de datos y se evaluó su funcionamiento. **Resultados.** Se logró fusionar correctamente los datos y se evaluó el método mediante la raíz del error cuadrático medio (RMSE), mostrando valores de efectividad por encima de 98% para la temperatura, 90% para la humedad y 99% para densidad de polvo. **Conclusión.** Sin duda, los valores fusionados pueden ser útiles, por ejemplo, para el análisis del confort térmico de los usuarios en ambientes cerrados, donde existen tasas de ventilación mínimas y se necesita una calidad de aire interior adecuada. Además, el análisis de variables físicas es útil para asegurar ambientes saludables y cómodos para los usuarios mientras realizan sus actividades cotidianas.

Índice general

Contexto	4
Contexto de la investigación	4
Problema de investigación	5
Objetivos general y específicos	6
Objetivo general	6
Objetivos específicos	6
Justificación	6
Organización del documento	7
Marco teórico	8
Fusión de datos	8
Origen de la fusión de datos	10
Idea Original de fusión de datos	10
Modelo de fusión de datos	10
Redes de sensores	11
Trabajos relacionados	14
Método de Solución	16
Instalación de la red de sensores	16
Recolección de datos	18
Fusión de datos	19
Método de Monte Carlo	19
Función utilizada	19
Evaluación del método de fusión	20
Resultados Obtenidos	21
Temperatura	21
Humedad	21
Densidad de Polvo	22
Conclusiones	24
Anexo A	25
Anexo B	42
Referencias	48

1. Contexto

1.1. Contexto de la investigación

La fusión de datos es la acción de integrar fuentes diversas para ser analizadas de acuerdo a un contexto determinado (Kumar *et al.*, 2007). Actualmente, la fusión se emplea en la integración de datos de fuentes variadas para hacer detecciones y clasificaciones de actividades en hogares inteligentes (Rodríguez *et al.*, 2015), ambientes virtuales inmersivos (Surie *et al.*, 2007), interfaces tangibles (Branton *et al.*, 2013), escritorios inteligentes (Yang *et al.*, 2013), Internet de las cosas (Cecchinell *et al.*, 2014), e incluso integración de datos cuantitativos y cualitativos, como la experiencia de expertos (Bogaert y Gengler, 2017).

El contexto es el entorno físico, emocional y social en el que se encuentra inmerso el usuario y que le dan significado, sentido y valor a las acciones o actividades que se realizan a su alrededor (Gellersen *et al.*, 2002). Esta característica de analizar el contexto del usuario se logra a través de sistemas conscientes de contexto (Khattak *et al.*, 2014), que adquieren información de su entorno, razonan y cambian el comportamiento del sistema para darle al usuario servicios que le sean de utilidad en la realización de sus actividades cotidianas (Bernardos *et al.*, 2008).

Para hacer esta adquisición, razonamiento y reacción se necesita analizar diferentes fuentes de datos. Mediante la fusión y análisis de estas fuentes de datos se puede caracterizar una entidad, persona, lugar u objeto, considerada relevante para la interacción entre el usuario y un sistema (Khattak *et al.*, 2014). Por lo que, en la actualidad los esfuerzos de investigación en los sistemas conscientes de contexto involucran el proceso de detección, adquisición, representación, fusión y razonamiento basado en el contexto adquirido.

Sin embargo, un problema creciente y crítico en los sistemas conscientes del contexto es cuando los datos surgen de fuentes de naturaleza diversa y en diferentes marcas de tiempo (Khattak *et al.*, 2014), como sensores, perfiles de usuario e interacciones sociales. Este problema, originado por la heterogeneidad de los datos, hace que se tengan datos en conflicto y desorden, conocido como inconsistencia, la cual es producida por el ruido ambiental, errores de calibración, fallas permanentes, entre otros factores. Estos datos en conflicto y desorden deben ser tratados para evitar resultados erróneos al ser fusionados con datos correctos.

1.2. Problema de investigación

En la actualidad, una de las áreas que mayor impulso ha dado a la fusión de datos es la minería y ciencia de datos, donde se emplean métodos para extraer, transformar y cargar fuentes de datos como parte del proceso de ingeniería de datos, previo a la analítica de estos (Jagadish *et al.*, 2014). Asimismo, Internet de las cosas (Internet of Things o IoT, por sus siglas en inglés) es un campo que también se ha beneficiado del concepto de fusión de datos, debido a que este es una tecnología que depende de objetos interconectados, capaces de comunicarse entre ellos y que colectan datos acerca de un determinado contexto (Cecchinell *et al.*, 2014). Otra de las áreas de creciente interés en el uso de la fusión de datos es la de los vehículos autónomos, donde los sistemas de navegación deben considerar el flujo de datos provenientes de diversas fuentes (sensores, cámaras y bases de datos) para tomar decisiones sobre el comportamiento del vehículo.

En general, el interés por la fusión de datos es cada vez mayor debido a la creciente incorporación de sensores en los dispositivos y el desarrollo de sistemas conscientes del contexto. El objetivo es tener información útil como apoyo para el proceso de la toma de decisiones sobre un determinado suceso, objeto o acción (Bogaert y Gengler, 2017), por ejemplo, en un sistema de salud consciente del contexto se necesita representar formalmente el contexto capturado usando sensores para monitorear el comportamiento diario de los pacientes (Khattak *et al.*, 2014). En estos sistemas los datos capturados son valores obtenidos en un determinado periodo, a esto se conoce como flujo de datos (Voida *et al.*, 2014). Este flujo de datos es procesado para identificar información de interés de los usuarios o del ambiente para hacer razonamientos (inferencias) impulsados por datos, por ejemplo, analizar el cambio de un comportamiento a través del tiempo.

Por lo tanto, una de las principales ventajas del uso de flujo de datos, como insumo en el proceso de fusión, es la flexibilidad del número, tipo y combinación de fuentes con base en el alcance del análisis que se pretende realizar. Esto es, se puede enfocar la recolección y análisis de los datos en un solo usuario, en un grupo de usuarios o en el entorno. De esta forma, mediante la fusión de datos de fuentes físicas, como sensores, se pueden formar configuraciones dinámicas de información, con el propósito de mejorar la inferencia del contexto.

Sin duda, la fusión de datos es clave en los sistemas con diversas fuentes de datos, las cuales debido a su naturaleza (tipo), variedad de formatos y velocidades de muestreo representan desafíos de investigación que necesitan ser estudiados, como (Khaleghi *et al.*, 2013): inconsistencia y correlación de datos. La inconsistencia de datos constituye uno de los problemas crecientes en la fusión de flujos de datos debido al conflicto y desorden en los datos ocasionado por el ruido ambiental, fallas permanentes, picos de corta duración en las

mediciones, o fallas que se desarrollan lentamente sin ser detectadas. Mientras que la correlación de datos es un tipo de problema causado por la duplicidad de datos en configuraciones distribuidas, donde algunos sensores obtienen los mismos datos debido a rutas cíclicas en el flujo de datos.

1.3. Objetivos general y específicos

1.3.1. Objetivo general

- Proponer un método para la fusión de datos de variables físicas, obtenidas a través de una red de sensores inalámbrica, medidas en un ambiente cerrado de ocupación humana.

1.3.2. Objetivos específicos

- Diseñar e implementar una red de sensores inalámbrica para la recolección de datos de variables físicas en un ambiente cerrado.
- Almacenar, visualizar y analizar los datos obtenidos a través de una plataforma en la nube.
- Diseñar un método de fusión de datos capaz de unificar las mediciones de las variables físicas, previamente obtenidas.
- Validar la eficiencia del método de fusión de datos.

1.4. Justificación

Un aspecto fundamental en la fusión de datos es la capacidad que deben tener los métodos de fusión para tratar la inconsistencia de datos provenientes de diversas fuentes (Khaleghi *et al.*, 2013). Esto debido a que la fusión de datos no es un proceso estático, sino dinámico, donde los datos que provienen de sensores pueden ser afectados por errores en las mediciones provocados por situaciones inesperadas, como: fallas en la medición, mediciones erróneas diferentes escalas de medición, o diferentes rangos de precisión (Guerriero *et al.*, 2016). Estos datos deben ser tratados para evitar errores durante el proceso de fusión, con el objetivo de tener una vista de datos precisa y unificada del objeto de estudio. Es de importancia tratar el conflicto, ya que, si solo es evitado o ignorado, los datos resultantes del proceso de fusión podrían no representar la situación actual en la que los datos fueron generados.

Así, la principal ventaja de fusionar datos es obtener un panorama global, unificado, coherente y preciso de un objeto, fenómeno, actividad que está siendo observado. Una de las

motivaciones de esta fusión es obtener datos que puedan servir de base para el análisis e identificación de patrones de comportamiento del usuario y del ambiente, con el objetivo de definir acciones y mecanismos adecuados a la situación, esto como apoyo para la inferencia de contextos (Lahat *et al.*, 2015).

En este sentido, en este documento, como opción de titulación por actividad de investigación, se describe el desarrollo de un método de fusión de datos de variables físicas, medidas a través de una red de sensores inalámbrica. Esta propuesta representa un importante campo de acción como base para el razonamiento en sistemas conscientes de contexto, por ejemplo, para el análisis del entorno con el fin de aprovechar el uso racional de la energía eléctrica y proporcionar un ambiente de trabajo cómodo para los usuarios.

1.5. Organización del documento

El documento está organizado de la siguiente manera, el Capítulo 2 presenta los fundamentos de la fusión de datos, se mencionan algunas de las contribuciones más significativas de la fusión de datos, se discuten sus aplicaciones, el uso de redes de sensores inalámbricos y los trabajos relacionados. El Capítulo 3 describe el método establecido como propuesta solución. El Capítulo 4 presenta los resultados obtenidos, basados en un ejemplo de aplicación, y el Capítulo 5 resume algunas conclusiones y trabajos futuros.

Se presenta además dos anexos, en el que se incluye información relacionada sobre el trabajo de investigación realizado. En el Anexo A se presenta el artículo de investigación publicado en la revista indizada *Research in Computing Science* (www.rcs.cic.ipn.mx/2020_149_11), cuyo título es ‘A Data Fusion Approach of Physical Variables Measured through a Wireless Sensor Network’. En el Anexo B se presenta el código del método de Monte Carlo utilizado para la fusión de datos de las variables temperatura, humedad y densidad de polvo; programados en Matlab (MATrix LABoratory).

2. Marco teórico

2.1. Fusión de datos

La fusión de datos es un campo de conocimiento usado para integrar datos producidos por uno o más sensores, o fuentes de datos para realizar un análisis y obtener una mejor estimación de la cantidad que se está midiendo ante la incertidumbre de observar un fenómeno físico (Wah-*Ng et al.*, 2006). Por otro lado, han surgido varias definiciones en la fusión de datos. Una de estas señala que es un proceso multinivel y multiestado para la detección, asociación, correlación, estimación y combinación de datos de una o más fuentes de datos para lograr estimaciones y evaluaciones completas (Blasch *et al.*, 2013).

Para tratar la inconsistencia de los datos hay diversos métodos que pueden ser utilizados, como vecinos cercanos, filtro del Kalman, estimaciones Bayesianas, simulación de Monte Carlo, entre otros:

- *Vecinos cercanos.* Se encarga de agrupar los valores que están unos cerca de otros, y presenten valores similares. La distancia de estos valores depende del límite propuesto, o de la distancia euclidiana. Una variante de vecinos cercanos es K-medias que busca agrupar los datos en un número de clústeres deseados, se asigna el valor central de cada clúster, y se recalculan los centros de forma iterativa. La desventaja de este método es conocer el número óptimo de clústeres iniciales.
- *Filtro de Kalman.* Es un modelo lineal recursivo que asigna un valor a un estado continuo en el tiempo. Se basa en observaciones periódicas de ese mismo estado para crear una estimación. Es un método que posee características que lo hacen útil para tratar la fusión de datos de múltiples sensores, ya que se basa en una distribución Gaussiana. Además, la medición de incertidumbre en los sensores es constante.
- *Estimación Bayesiana.* La inferencia Bayesiana es un método de fusión de datos basado en el teorema de Bayes que calcula la distribución de probabilidad posterior de un vector n-dimensional 'X' después de haber observado un evento 'Z' (Muñoz-Benítez *et al.*, 2018). La información probabilística está descrita por una función de densidad de probabilidad conocida como función verosimilitud: $p(Z | X)$.
- *Simulación de Monte Carlo.* Es un método ampliamente utilizado en el análisis probabilístico de sistemas. Este método basa su funcionamiento en una

experimentación numérica para obtener estadísticas de las variables de salida, dado un conjunto de variables de entrada (Siciliano y Khatib, 2016). En cada experimento, los valores de entrada son muestreados, basados en sus distribuciones, y las variables de salida son calculadas usando el sistema.

Por otro lado, la fusión de datos ha tenido importantes avances en aplicaciones militares como: reconocimiento automático de objetivos, navegación de vehículos autónomos, censo remoto, sistemas de identificación y vigilancia, entre otros (Llinas y Hall, 1998). Además, los datos obtenidos de los sensores pueden ser fusionados simultáneamente, conocida como fusión centralizada, o secuencialmente, conocida como fusión descentralizada.

En la actualidad, los métodos de fusión de datos son utilizados en diversas aplicaciones de análisis avanzado de datos. Uno de estos es el análisis de negocios, donde se analizan datos para mejorar procesos internos, identificar tendencias, descubrir oportunidades de mejora y reducir costos. Otro es en robots móviles, donde se utilizan diversos tipos de sensores para obtener una mejor noción de los objetivos, detección de objetos y navegación en un entorno de trabajo (Salina y Malathi, 2014).

La fusión de datos también se emplea para hacer detecciones y clasificaciones de actividades en hogares inteligentes para ayudar a adultos mayores en caso de accidentes en el hogar (Rodríguez *et al.*, 2015). Mientras que en el área de Internet de las Cosas, la fusión de datos puede ser usada para medir ocupación de espacios en estacionamientos y edificios con aire acondicionado con el objetivo de ahorrar tiempo o energía eléctrica (Cecchinell *et al.*, 2014).

2.1.1. Origen de la fusión de datos

La fusión de datos es un subcampo de la fusión de información, también conocido como integración de información. El objetivo es integrar información proveniente de distintas fuentes. Mientras que la fusión de información es utilizada en la comunicación entre dos o más entidades, haciendo que la información sea fusionada, dando como resultado una nueva información. Además, es usada también para eliminar la redundancia en la información.

Debido a la necesidad en la comunidad científica de estandarizar los términos y desarrollar un modelo general que sirviera en varias áreas, se dio la definición más antigua de fusión de datos, proporcionada por el Grupo de Fusión de Datos JDL en 1987 (Steinberg *et al.*, 1999): “la fusión de datos es el proceso de lidiar con la asociación, correlación y combinación de datos e información de una o diversas fuentes para lograr estimaciones

oportunas de situaciones o amenazas”. Estas estimaciones son hechas a partir de información refinada y reducida con el propósito de optimizar el proceso de toma de decisiones.

Por lo tanto, la fusión de datos proporciona una formalización para la integración de datos, de tal manera que los resultados ayuden en el proceso de toma de decisiones sobre un determinado evento, objeto o acción (Durrant-Whyte y Henderson, 2001).

2.1.2. Idea original de fusión de datos

La idea original de la fusión de datos es descubrir que tan confiable es la medición de un fenómeno físico con el objetivo de mejorar el entendimiento y la respuesta a tal fenómeno. Así, la información que es recibida de una sola fuente de información puede contener errores. Esto puede ser ocasionado por un error en la fuente de datos, en el proceso de transferencia, o en el hecho de estar limitado a una o más fuentes de información.

Cuando no se puede confiar en la información recibida es necesario hacerse algunos cuestionamientos respecto al contexto estudiado, como: i) ¿para quién es el mensaje?, ii) ¿para qué se desea transmitir el mensaje?, iii) ¿a dónde o a quién se debe transmitir el mensaje?, y d) ¿el mensaje es confiable?. Estas preguntas se deben a que se busca entender con claridad la intención de un mensaje.

La respuesta a la pregunta ¿para quién?, es el receptor del mensaje. La respuesta a la pregunta ¿para qué?, es el propósito (para entablar la comunicación). La respuesta a la pregunta ¿dónde?, es la ubicación del receptor. Por ende, esto significa tener confianza en una medición cuando se trabaja con sensores. En este sentido, una forma de obtener una medición de confianza es obteniendo la información de múltiples fuentes y fusionarlas.

2.2. Modelo de fusión de datos

El modelo JDL, propuesto por Joint Director of Laboratories (JDL), que hoy en día se llama Data Fusion Group, fue creado debido a la necesidad de estandarizar sistemas de fusión de datos de forma efectiva, reutilizando recursos. Este modelo consiste en niveles de abstracción (Liggins *et al.*, 2008), para detección y extracción de características (nivel 0), identificación de objetos (nivel 1), relaciones (nivel 2), impactos (nivel 3), y respuestas (nivel 4):

- *Nivel 0.* Definido para la detección y extracción de características de una imagen y estimación de estados observables a un bajo nivel. Durante la extracción, la cantidad

de datos es reducida manteniendo información útil para los niveles superiores, es decir, se busca refinar las características para ayudar en la identificación del objeto.

- *Nivel 1.* Definido para la estimación y predicción de los estados de una entidad con base en la asociación entre una observación y una hipótesis. Para esto, se utilizan los datos procesados en el nivel anterior con el propósito de identificar y clasificar el objeto, así como definir su estado y orientación. Este nivel transforma la información de bajo nivel, procesada previamente, en estructuras de datos.
- *Nivel 2.* Definido para identificar las relaciones entre elementos con base en la observación de eventos y datos recolectados en los niveles previos. De esta forma, se establecen las relaciones existentes entre los objetos, como proximidad, comunicación, rol, por mencionar algunos. Estas relaciones son evaluadas para determinar la importancia de los objetos en un ambiente determinado e identificar actividades o eventos para la definición de patrones.
- *Nivel 3.* Definido para evaluar el impacto de las actividades o eventos identificados en el nivel anterior. El impacto es evaluado para identificar posibles riesgos, vulnerabilidades y oportunidades operacionales, por lo que, se busca predecir el resultado de las actividades o eventos identificados previamente.
- *Nivel 4.* Definido para la asignación de tareas o acciones según los recursos disponibles. Se encarga de gestionar, de manera eficiente, las respuestas a los eventos, relaciones, y objetos identificados y evaluados en los niveles previos. De tal manera que se ejecuten respuestas óptimas con el menor uso de recursos.

A través de estos niveles se pueden abstraer e identificar entidades, como objetos, personas o eventos, para establecer sus relaciones con el ambiente, el usuario, e incluso con otras entidades. El propósito es tener aplicaciones, basadas en fusión de datos, que reaccionen de forma eficiente como sea requerido.

2.3. Redes de sensores

Una red de sensores es una red interconectada de dispositivos sensoriales que cooperan entre ellos para medir y controlar un determinado contexto físico. Los sensores trabajan dentro de un campo de acción para capturar y transformar las señales en datos, que se envían en tiempo real a una unidad central, permitiendo la interacción entre el usuario y una máquina, como Internet o una base de datos (Akyildiz *et al.*, 2002). Uno de los principales retos en el diseño

de una red de sensores es desarrollar sensores que sean accesibles, usen poca energía y puedan ser desplegados con facilidad.

La arquitectura de una red de sensores se basa en la forma en que estos sensores se relacionan entre sí y depende de su aplicación específica. Por ejemplo, en Bokareva *et al.* (2006) se define una arquitectura de rastreo de objetivos, dividida en tres fases: i) la primera se compone de los sensores que obtienen datos de un ambiente físico, ii) en la segunda se tiene un algoritmo que agrega los datos, y iii) en la tercera se tiene la fusión de datos. La salida de la fusión de datos es el rastreo del objeto.

Por otro lado, la organización física de una red de sensores puede ser distribuida, centralizada y descentralizada.

- *Distribuida.* Donde no hay nodo central. Los nodos se encuentran físicamente separados, generalmente en una amplia área y puede existir cooperación entre estos para realizar sus tareas. Estos nodos son buenos para prevenir un solo punto de fallas.

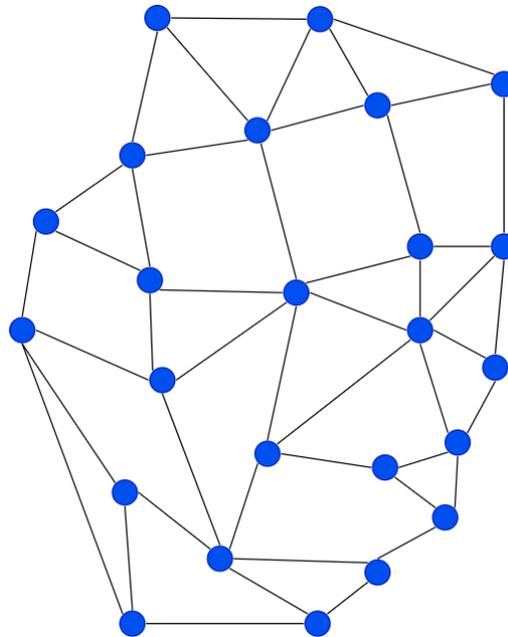


Figura 1. Diagrama de una red de sensores distribuida.

- *Centralizada.* Cada nodo trabaja por sí solo y los datos son enviados a un nodo central. Este nodo central es el responsable de filtrar datos y realizar los cálculos necesarios para hacer una estimación del problema y tomar una decisión. El problema de esta arquitectura es que si el nodo central falla la integración de información también lo hace.

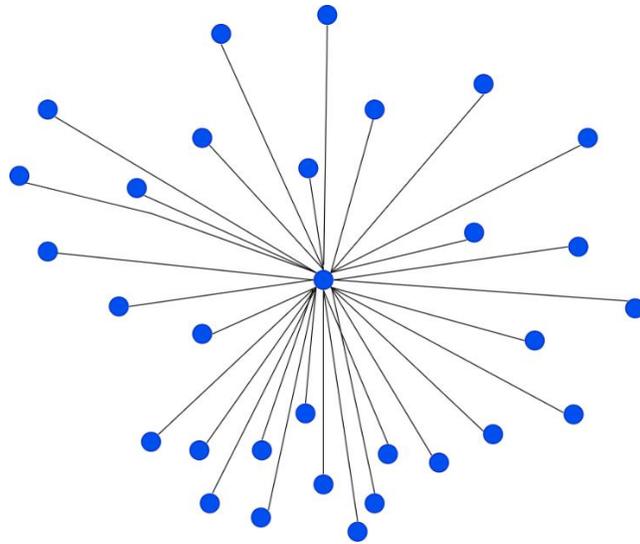


Figura 2. Diagrama de una red de sensores centralizada.

- *Descentralizada.* Se distribuye la red en varios puntos, donde algunos de estos sirven como unidad central, si alguna parte de la red cae, la red sigue funcionando, sin embargo, no todas las unidades locales son iguales en cuanto a carga de trabajo o capacidades de procesamiento.

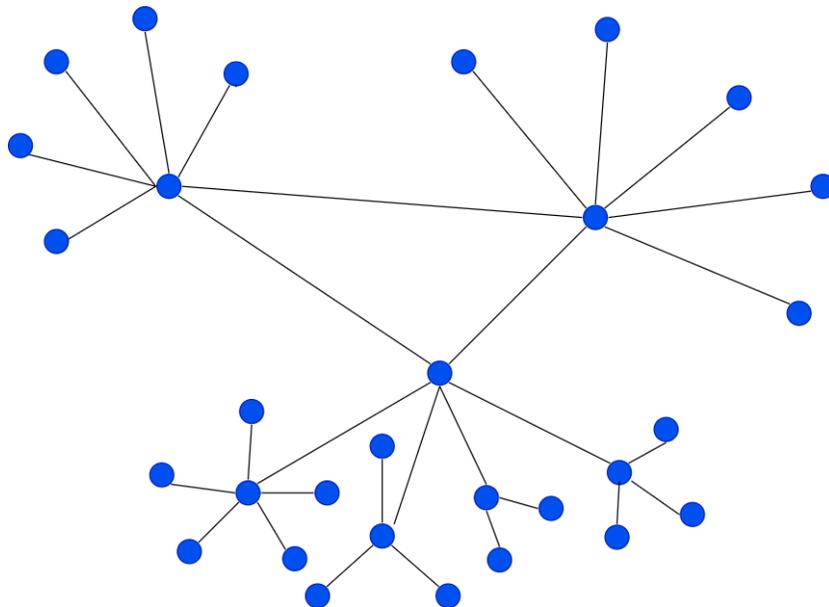


Figura 3. Diagrama de una red de sensores descentralizada.

En este sentido, hoy en día, dada la aparición de circuitos electrónicos cada vez más pequeños, con alto nivel de procesamiento y de menor costo, se han desarrollado dispositivos para la detección y medida de variables físicas. Así, a través de estos dispositivos, con capacidades de comunicación, bajo costo y elevada duración, se puede explotar su uso en el

despliegue de redes inalámbricas, sin necesidad de mantenimiento, capaces de obtener información del entorno y enviarla a un centro de almacenamiento, lo que ofrece una variedad de aplicaciones como el monitoreo del ambiente en determinadas áreas de interés.

2.4. Trabajos relacionados

En la actualidad, dada la creación, recolección y almacenamiento de datos a gran escala, que varían en complejidad, existe un creciente interés por la fusión de datos y el análisis de estos (Chen *et al.*, 2014). Decisiones que antes eran basadas en presunciones, ahora se pueden modelar características para obtener una descripción, predicción o incluso una prescripción para tomar una determinada decisión con base en un análisis previo.

No obstante, un problema creciente en la fusión de datos es lidiar con la inconsistencia de los datos. Para esto, existen varios métodos que se pueden utilizar, como enviar el mismo paquete de datos por segunda vez. Una vez enviados, se comparan las similitudes de los dos paquetes y se eliminan los valores atípicos. Otra opción es comparar mediciones de datos anteriores y luego dar un cierto nivel de confianza a los sensores que se utilizaron (Kumar y Garg, 2009). Otro método es asignar una serie de pesos jerárquicamente a los sensores (Frikha y Moalla, 2015). La Tabla 1 resume algunos trabajos identificados, como parte de la revisión de la literatura.

Tabla 1. Trabajos relacionados.

Autor	Descripción	Método usado	Limitaciones
Motro <i>et al.</i> (2006)	Es un sistema de integración de fuentes de datos llamado Fusionplex,	El proceso consiste en encontrar inconsistencias en las tuplas de datos. Es decir, se detectan y eliminan las inconsistencias acuerdo con criterios establecidos.	Alto costo computacional para resolver conflictos en aplicaciones comerciales.
Kumar y Garg (2009)	Detecta la inconsistencia de los datos en los sensores antes de enviarlos.	Utiliza una red neuronal para detectar la inconsistencia de los datos a través de su varianza.	No analiza la inconsistencia de los datos.

Frikha y Moalla (2015)	Asigna una jerarquía a los sensores para eliminar la subjetividad y reducir la inconsistencia.	Los sensores se compararon en pares empleando matrices. Se añaden los pesos de los sensores. Cuanto mayor sea la jerarquía, mayor será el peso.	Se requiere definir los criterios para asignar las jerarquías iniciales.
Bakr y Lee (2017)	Se presentan tres enfoques de fusión de datos para tratar la inconsistencia: a) basado en modelos, b) basado en redundancia, y c) basado en fusión.	Se describe el tratamiento de la inconsistencia de los datos de diversas fuentes.	Carece de un proceso de fusión de datos de múltiples fuentes.

En este sentido, debido al prometedor futuro de las redes de sensores inalámbricos, existe un interés natural por adoptar aplicaciones de fusión de datos con este tipo de tecnología para monitorear diferentes fenómenos y eventos de manera eficiente.

3. Método de solución

Para el logro del objetivo, el método de trabajo definido para la fusión de datos de variables físicas, medido en un ambiente cerrado de ocupación humana, se dividió en cuatro etapas: a) instalación de la red de sensores, b) recolección de datos, c) fusión de datos y d) evaluación del método de fusión.

3.1. Instalación de la red de sensores

En esta etapa se instalaron dos nodos sensores en los extremos de una habitación. Inicialmente, antes de la pandemia por COVID-19, estos nodos fueron instalados en uno de los ambientes del Departamento de Computación de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México. Sin embargo, ante las medidas sanitarias por COVID-19, definidas en marzo de 2020 a nivel universitario, se decidió trasladar los sensores a un domicilio particular y hacer las pruebas en una habitación.

En este sentido, la placa NodeMcu (ESP8266) fue programada para enviar datos a la plataforma ThingSpeak, que permite almacenar, visualizar y analizar datos en tiempo real, utilizando servicios en la nube para aplicaciones de Internet de las Cosas (IoT). La plataforma cuenta con una interfaz de programación de aplicaciones (API) para trabajar con los nodos a través de Internet. Esto permite acceder a los datos recopilados desde cualquier parte del mundo, utilizando cualquier dispositivo con una conexión a Internet. La Figura 4 muestra el diagrama de un nodo y la Tabla 2 resume las principales características de la placa NodeMcu, los valores requeridos de voltaje, velocidad y tamaño.

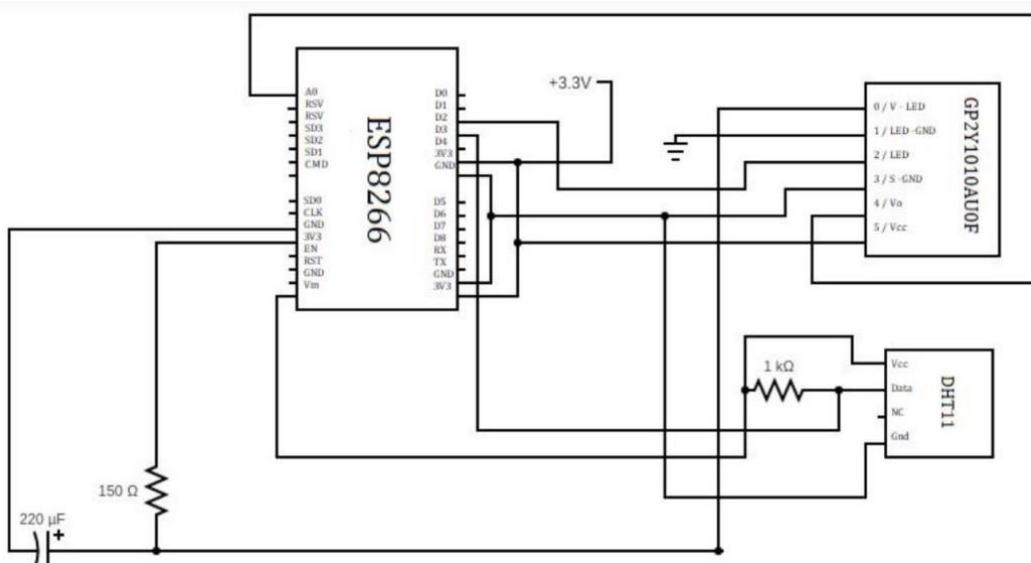


Figura 4. Diagrama de conexión de un nodo.

Tabla 2. Características de la placa NodeMcu.

Ítem	Características	Valor
1	Fuente de poder	5 V
2	Voltaje de salida	3.3 V
3	Velocidad de Procesamiento	80 - 160 MHz
4	Tamaño	29.1 x 63.3 mm

Aunado a la placa NodeMcu, se utilizó el sensor DHT11, que incluye un componente de medición de humedad y temperatura, el cual se conecta a un microcontrolador de 8 bits para garantizar una calidad y respuesta rápida utilizando bajo voltaje. La Tabla 3 resume las principales características de este sensor.

Tabla 3. Características del sensor DHT11.

Ítem	Características	Valor
1	Fuente de poder	3.3 – 5 V
2	Salida de voltaje	3.3 V
3	Rango de humedad	20 – 90% con 5% de precisión
4	Rango de temperatura	0 – 50°C con $\pm 2^\circ\text{C}$ de precisión
5	Tamaño	14 x 18 x 5.5 mm

Otro de los sensores utilizados fue GP2Y1010AU0F, el cual sirve para detectar la luz reflejada de las partículas de polvo y humo. Este tipo de sensor puede distinguir entre

partículas de polvo y humo por medio de un patrón de pulso en el voltaje de salida. La Tabla 4 presenta algunas de sus características importantes.

Tabla 4. Características del sensor GP2Y1010AU0F.

Ítem	Características	Valor
1	Fuente de poder	5 V
2	Consumo de corriente	Icc: MAX 20 mA
3	Ciclo de pulso	10 ± 1ms
4	Tamaño	46.0 x 30.0 x 17.6 mm

3.2. Recolección de datos

Para el envío de los datos a los canales de ThingSpeak y mantener su integridad, se utilizó la función ‘thingSpeakRead’, disponible en la plataforma, que importa los valores no nulos. Así, para el envío de los datos provenientes de los dos nodos, se crearon un canal para cada uno, mediante los cuales se almacenaron los datos y se otorgaron accesos para que fueran leídos por los involucrados en el proyecto. Los datos se recopilaban en un período de cinco semanas, de mayo a junio de 2020 y fueron almacenados en archivos de valores separados por coma (csv), los cuales se descargaron a través de los dos canales establecidos:

- Nodo 1: https://thingspeak.com/channels/1010164/private_show
- Nodo 2: https://thingspeak.com/channels/1010156/private_show

La Tabla 5 muestra, a modo de ejemplo, un extracto de los datos capturados, donde las variables físicas capturadas por los sensores fueron: temperatura en grados centígrados (°C), humedad en porcentaje (%), y densidad de polvo en miligramos de contaminante por cada metro cúbico (mg/m³).

Tabla 5. Fragmento de los datos recolectados a través de los sensores.

Fecha	Temperatura °C	Humedad %	Densidad de Polvo mg/m ³
2020-05-30 19:54:38	28.9	67	0.02
2020-05-30 19:59:38	29.1	69	0.01
2020-05-30 20:04:39	29	71	0.02
...
2020-06-23 09:37:12	32.1	64	0.01
2020-06-23 09:42:13	32.2	65	0.01
2020-06-23 00:47:13	32.2	66	0.02

3.3. Fusión de datos

Para la fusión de datos se utilizó el método de Monte Carlo, con el fin de verificar la variación en los datos obtenidos mediante los nodos sensores. Este método fue elegido por su capacidad para operar numéricamente sistemas complejos, manteniendo la relación de entrada y salida, con un cierto grado de efectividad. Además, a través de este método es posible identificar posibles conflictos generados en los datos, ya que, si se ignora esta situación, los datos resultantes podrían representar, de manera errónea, los datos reales obtenidos en la experimentación. Posteriormente, los datos se fusionaron para cada una de las variables recolectadas, es decir, temperatura, humedad y densidad de polvo.

3.3.1. Método de Monte Carlo

Es un método no determinista, utilizada para aproximar estimaciones numéricas a una gran variedad de problemas matemáticos. Este método es nombrado en referencia al Casino de Monte Carlo (Mónaco), por ser la capital de los juegos de azar, puesto que la ruleta es un generador de números aleatorios (Illana, 2013). Basa su funcionamiento en la simulación, la cual consiste en repetir las características y comportamientos de un sistema real. El objetivo es imitar el comportamiento de las variables reales para que, en la medida de lo posible, se analicen y pronostiquen valores futuros simples o complejos.

Actualmente, este método es una parte fundamental de los algoritmos de Ray Tracing para la generación de imágenes 3D. El método puede variar dependiendo de la aplicación, pero en general, sigue los siguientes pasos: i) se define el dominio de las entradas de la función; ii) se elige una distribución de probabilidad y se generan valores aleatorios; iii) se evalúan las entradas en la función, y iv) se reúnen los valores calculados.

3.3.2. Función utilizada

Al ser un algoritmo de análisis probabilístico, la posibilidad de error está dada por el número de errores dividido por el número de veces que se realiza la prueba. Siguiendo el teorema del límite central, Monte Carlo tiene un error absoluto dado por:

$$\frac{1}{\sqrt{N}}$$

Esto indica que, si se aumenta el número de pruebas, la probabilidad del error tiende a cero (Illana, 2013). Una ventaja de la simulación Monte Carlo es que puede dar una buena aproximación del valor exacto. De esta manera, este método se puede utilizar para medir la fiabilidad en una red de sensores (Yang *et al.*, 2011). La ecuación utilizada fue la siguiente:

$$\langle b - a \rangle \frac{1}{N} \sum_{n=1}^{\infty} f(x_i)$$

donde:

- $\langle b - a \rangle$ son los intervalos de la función.
- N es el número de pruebas.
- x_i es el valor aleatorio que es evaluado en la función.

3.4. Evaluación del método de fusión

Como método de evaluación de la fusión de datos se utilizó la raíz del error cuadrático medio (RMSE, por sus siglas en inglés). A través de esta medición, se calcula la diferencia (error) entre los valores estimados (pronosticados) por el método de Monte Carlo y los valores capturados por los sensores. La ecuación utilizada fue la siguiente:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^n \left(\frac{d_i - f_i}{\sigma_i} \right)^2}$$

Normalmente, se utiliza la raíz del error cuadrático medio para dar una estimación en términos de la calidad del pronóstico.

4. Resultados obtenidos

Los resultados de la fusión de datos para cada una de las variables analizadas se muestran a continuación.

4.1. Temperatura

Se trazaron las distribuciones de probabilidad de los valores obtenidos para la temperatura, así como la fusión de datos. La Figura 5 muestra las mediciones de los sensores 1 (azul) y 2 (verde), que están cerca del percentil 50 de cada uno. Mientras que los valores de la fusión de datos (rojo) se encuentran entre las mediciones de los dos sensores, con una ligera similitud con los valores obtenidos mediante el sensor 2.

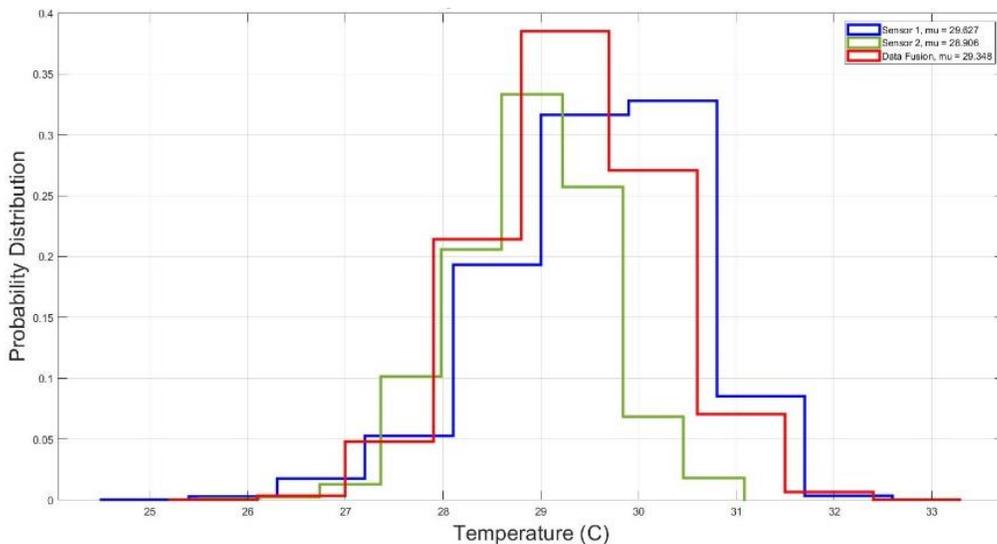


Figura 5. Distribución de probabilidad de fusión de datos de temperatura.

4.2. Humedad

Se trazaron las distribuciones de probabilidad de los porcentajes de humedad, así como la fusión de datos (Figura 6). Se encontró que los valores más probables para el sensor 1 (azul) y el sensor 2 (verde) estaban dentro de los valores de la fusión de datos (rojo). Las mediciones del sensor 1 (azul) tienen un comportamiento promedio en relación con sensor 2 y la fusión de datos.

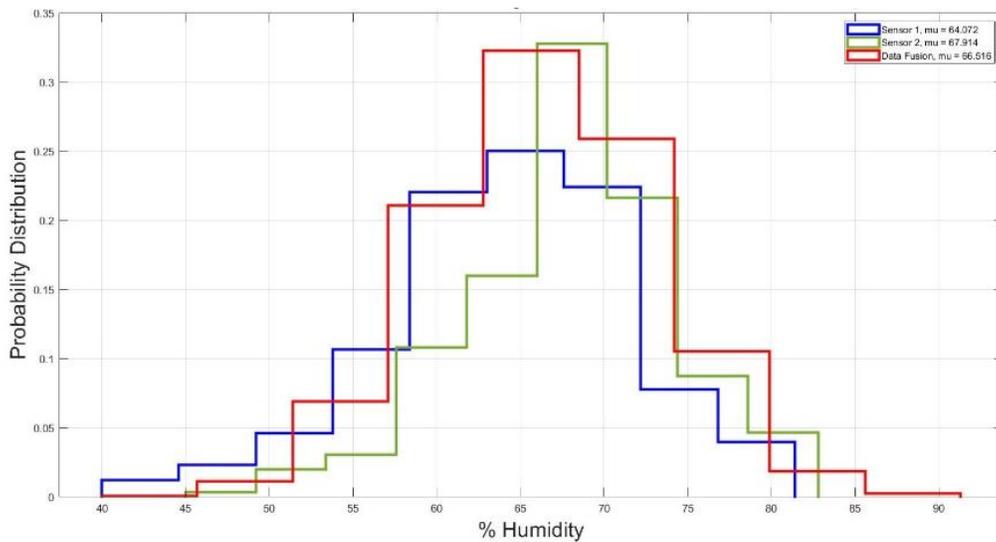


Figura 6. Distribución de probabilidad de fusión de datos de humedad.

4.3. Densidad de polvo

La Figura 7 muestra la distribución de probabilidad de los porcentajes de densidad de polvo medidos por los dos sensores, así como los valores de la fusión de datos. Se observó que los valores del sensor 2 (verde) eran inusuales, ya que están sesgados hacia la derecha. Esto puede deberse a que el sensor 2 se encuentra encima de una estantería y a que la cantidad de polvo que se levantó fue mayor debido a la limpieza realizada en esta parte de la habitación. Además, se observó que el sensor 1 (azul) presentaba menos variación, con una probabilidad de densidad inferior a 0.05, es decir, casi el 99%. Aunque la mayoría de los datos obtenidos en el sensor 1 variaron entre 0.00 y 0.05, se presentaron algunos casos de variabilidad entre 0.07 y 0.4; por lo que, se asume que el sensor funciona correctamente.

Esta diferencia de mediciones entre el sensor 1 y 2 se debe fundamentalmente a la ubicación en la habitación en los que fueron instalados, donde en uno de los lados casi no hay flujo de aire y en otro no se hicieron limpiezas continuas.

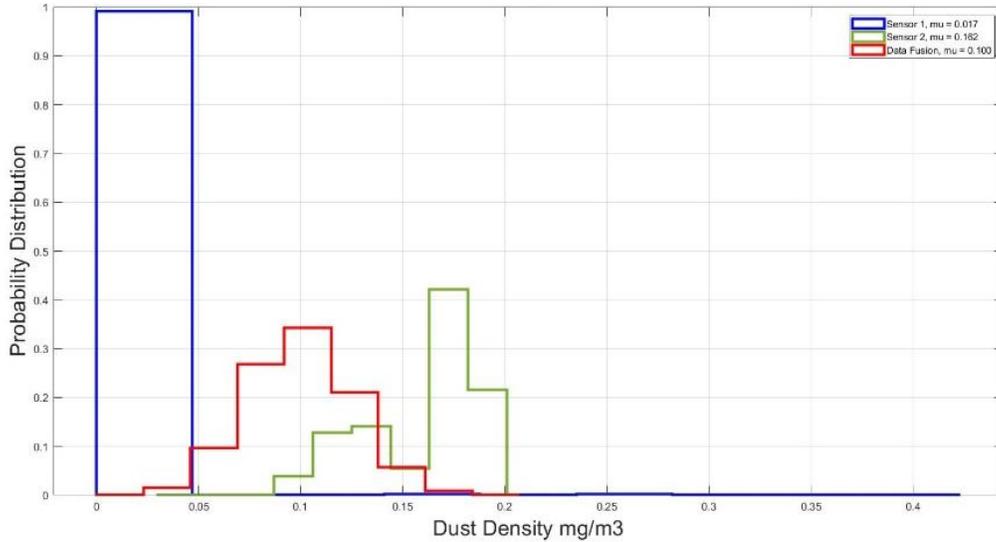


Figura 7. Distribución de probabilidad de fusión de datos de densidad de polvo.

Por otro lado, la Tabla 6 muestra los resultados obtenidos de las mediciones de la raíz del error cuadrático medio con respecto a la fusión de datos. Se alcanzaron bajos niveles de error en temperatura, con 1.38 y 1.25 unidades de error que se alejan del valor real; y densidad de polvo con 0.08 y 0.07 unidades de error que se alejan del valor real. Mientras que para la humedad se observaron mayores porcentajes de error, es decir, de 9.27 y 10.14 unidades. Esto puede deberse a que algunos días llueve y otros no. Cuando llueve, la humedad aumenta, esto es, hay una alta saturación, lo que puede dar valores inusuales.

Tabla 6. Estimación de la raíz cuadrada del error cuadrático medio.

Variable	Sensor	RMSE
Temperatura	Sensor 1	1.3892
	Sensor 2	1.2586
Humedad	Sensor 1	10.1487
	Sensor 2	9.2745
Densidad de polvo	Sensor 1	0.0895
	Sensor 2	0.0735

En consecuencia, el nivel de precisión promedio de fusión de datos para las mediciones de temperatura fue de 98.7%, para la humedad de 90.3%, y para la densidad de polvo de 99.9%. Este valor de 99.9% en los niveles de densidad de polvo se debe a que las mediciones del RMSE para los dos sensores de densidad de polvo fueron bajos, entre 0.00 y 0.05. Por otra parte, es importante mencionar que la precisión del método de Monte Carlo aumenta a medida que se realizan más simulaciones. Por lo tanto, el costo computacional también aumenta.

5. Conclusiones

Se implementó una red de sensores en un área de descanso y trabajo, dado el contexto actual por la enfermedad COVID-19, como es una habitación en un hogar, con el fin de obtener datos sobre temperatura, humedad y densidad de polvo. Para esto, se utilizó una tarjeta de desarrollo NodeMcu (ESP 8266) con capacidades de conexión Wi-Fi con el propósito de conectarse a la plataforma ThingSpeak para recopilar, visualizar y analizar datos de los sensores.

Se utilizaron dos sensores DHT11 para medir la temperatura y humedad, y se utilizaron dos sensores GP2Y1010AUOF para medir la densidad de polvo. Los datos obtenidos por los sensores se almacenaron en la nube a través de dos canales creados en ThingSpeak. Precisamente, mediante estos canales se pueden consultar los datos en tiempo real y ser descargados a través de la función 'Extraer datos'.

Los valores obtenidos para las variables de temperatura, humedad y densidad de polvo se fusionaron correctamente utilizando el método Monte Carlo. Los valores de los sensores y los valores obtenidos a través del método utilizado se compararon. Además, se alcanzaron bajos niveles de error, que se evaluaron mediante la raíz del error cuadrático medio, lo que permitió verificar la exactitud de los valores obtenidos.

Como trabajo futuro, se buscará analizar con base en los valores fusionados de temperatura, humedad y densidad de polvo, si el ambiente de observación, en este caso, una habitación, se encuentra dentro de la zona de confort térmico, definido según la norma ANSI/ASHRAE 55. Este tipo de análisis es útil para identificar las tasas mínimas de ventilación y otras medidas para proporcionar una calidad adecuada del aire interior en espacios ocupados por el ser humano.

Se considerará también ampliar la implementación con otros métodos, como redes neuronales artificiales y filtro de Kalman, con el propósito de comparar los resultados y obtener mejores datos fusionados.

Finalmente, el trabajo realizado implicó retos importantes, como el diseño e implementación de los nodos de la red de sensores inalámbrica; la gestión de datos en la nube a través de ThingSpeak; y el desarrollo del método de fusión de datos de variables físicas. Permitiendo así extender la visión de cómo desarrollar un proyecto y artículo de investigación y su aplicación a problemas reales.

Anexo A

En este apartado se presenta el artículo de investigación publicado en la revista Research in Computing Science, indizada en LatIndex, DBLP y Periodica.



Research in Computing Science

Series Editorial Board

Editors-in-Chief:

Grigori Sidorov, CIC-IPN, Mexico
Gerhard X. Ritter, University of Florida, USA
Jean Serra, Ecole des Mines de Paris, France
Ulises Cortés, UPC, Barcelona, Spain

Associate Editors:

Jesús Angulo, Ecole des Mines de Paris, France
Jihad El-Sana, Ben-Gurion Univ. of the Negev, Israel
Alexander Gelbukh, CIC-IPN, Mexico
Ioannis Kakadiaris, University of Houston, USA
Olga Kolesnikova, ESCOM-IPN, Mexico
Petros Maragos, Nat. Tech. Univ. of Athens, Greece
Julian Padget, University of Bath, UK
Mateo Valero, UPC, Barcelona, Spain
Rafael Guzmán, Univ. of Guanajuato, Mexico
Juan Manuel Torres Moreno, U. of Avignon, France

Editorial Coordination:

Alejandra Ramos Porras

Research in Computing Science, Año 19, Volumen 149, No. 11, noviembre de 2020, es una publicación mensual, editada por el Instituto Politécnico Nacional, a través del Centro de Investigación en Computación. Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, Ciudad de México, Tel. 57 29 60 00, ext. 56571. <https://www.rcs.cic.ipn.mx>. Editor responsable: Dr. Grigori Sidorov. Reserva de Derechos al Uso Exclusivo del Título No. 04-2019-082310242100-203. ISSN: en trámite, ambos otorgados por el Instituto Politécnico Nacional de Derecho de Autor. Responsable de la última actualización de este número: el Centro de Investigación en Computación, Dr. Grigori Sidorov, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738. Fecha de última modificación 01 de noviembre de 2020.

Las opiniones expresadas por los autores no necesariamente reflejan la postura del editor de la publicación.

Queda estrictamente prohibida la reproducción total o parcial de los contenidos e imágenes de la publicación sin previa autorización del Instituto Politécnico Nacional.

Research in Computing Science, year 19, Volume 149, No. 11, November 2020, is published monthly by the Center for Computing Research of IPN.

The opinions expressed by the authors does not necessarily reflect the editor's posture.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research of the IPN.

Volume 149(11)

Artificial Intelligence and Industry 4.0

**Juan Humberto Sossa Azuela
Maria Elena González Bañales
Edgar Gonzalo Cossio Franco (eds.)**



Instituto Politécnico Nacional
"La Técnica al Servicio de la Patria"



Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2020

ISSN: in process

Copyright © Instituto Politécnico Nacional 2020
Formerly ISSN: 1870-4069, 1665-9899.

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional “Adolfo López Mateos”, Zacatenco
07738, México D.F., México

<http://www.rcs.cic.ipn.mx>

<http://www.ipn.mx>

<http://www.cic.ipn.mx>

The editors and the publisher of this journal have made their best effort in preparing this special issue, but make no warranty of any kind, expressed or implied, with regard to the information contained in this volume.

All rights reserved. No part of this publication may be reproduced, stored on a retrieval system or transmitted, in any form or by any means, including electronic, mechanical, photocopying, recording, or otherwise, without prior permission of the Instituto Politécnico Nacional, except for personal or classroom use provided that copies bear the full citation notice provided on the first page of each paper.

Indexed in LATINDEX, DBLP and Periodica

Electronic edition

Editorial

In this special issue of the prestigious journal "Research in Computing Science", artificial intelligence and industry 4.0 works are presented, which were carefully selected under the triple review scheme by the technical committee made up of national and international experts in the field through the EasyChair platform.

The selection was based on their originality, impact, contribution and scientific rigor. The acceptance rate was 54% and the topics covered range from machine learning, deep learning, computer vision, big data, IoT, robotics, smart cities, among other topics that directly impact Industry 4.0.

The papers presented in this special issue, as well as their authors, are described below: 1) Frequency of Mind-Wandering in a Sustained Attention to Response Task: A Cognitive Model of Distraction, author Renzo Cuadra, 2) Traffic Prediction Architecture based on Machine Learning approach for Smart Cities, authors Marco Hernandez Inzunza, Luis Hernandez Robles, Miriam Carlos Mancilla & Emmanuel López Neri, 3) Utility-pole, nuts and cross-arm visual detection, for electric connections maintenance robot, authors José Luis Gomes Torres, Saúl Martínez Díaz, Iddaly Iris Méndez Gurrola, Alejandro Israel Barranco Gutierrez & Kyoichi Tatsuno, 4) Proposal for a control, monitoring and assistance system for optimization of energy resources at home, authors Leandro Espino Espino, Yotziri Paloma Pérez Rios & Edgar Gonzalo Cossio Franco, 5) Intelligent model to determine the software development effort through machine learning, authors Edgar Gonzalo Cossio Franco, Mariela Chávez Marcial, Emma Yesenia Rivera Ruíz, Yotziri Paloma Pérez Rios & Marco Julio Franco Mora, 6) Selection of Actions for Navigation of a Mobile Robot Based on Fuzzy Q-Learning, authors Elizabeth López Lozada, Elsa Rubio, Humberto Sossa & Victor H. Ponce Ponce, 7) Attribution of Authorship of Twitter Messages through Automatic Syntactic Analysis, authors Francisco Antonio Castillo Velázquez, José Luis Martínez Godoy, María del Consuelo Patricia Torres Falcon, Jonny Paul Zavala de Paz, Adela Becerra Chavez & José Amilcar Rizzo Sierra, 8) A data fusion approach of physical variables measured through a wireless sensor network, authors Thomas A. Arias Pelayo, Guillermo Molero Castillo, Everardo Bárcenas & Rocío Aldeco Pérez, 9) A Physical Variable Data Fusion Approach as part of Ambient Intelligence, authors Julio Muñoz Benítez, Guillermo Molero Castillo, Everardo Bárcenas, Rocío Aldeco Pérez & Alejandro Velázquez Mena, 10) Bio-inspired Computing in Software Testing: a Systematic Literature Review, authors Jethran Gomez San Gariel, Angel Sanchez Garcia & Karen Cortes Verdín, 11) Design and construction of an IoT Humidity Sensor made of PVDF for applications in the Crop Industry, authors Lizbeth Berenice Contreras Cariño & Carlos Omar González Morán, 12) Algorithm for counting models in FNC, authors Pedro Bello López, Meliza Contreras González & Miguel Rodríguez Hernández, 13) Performance evaluation of infrared, ultrasonic and vision sensors for vehicle counting, authors Juan José Lozano Barrón, Osslán Osiris Vergara Villegas, Salvador Noriega Morales & Vianey Guadalupe Cruz Sánchez, 14) On the algebrization of paraconsistent logic CG'3, authors Miguel Pérez Gaspar & Everardo Ismael Bárcenas Patiño, 15) A Systematic Mapping Study of Artificial Intelligence in Software Requirements, authors Alfonso Robles Aguilar, Jorge Octavio Ocharán Hernández & Angel Sanchez

Garcia, 16) Gasoline price detector based in artificial vision and cloud computing, authors Mircel Karel García Rodríguez, Osslán Osiris Vergara Villegas & Ivón Oristela Benitez Gonzalez, 17) Continuous retraining of an LSTM algorithm with a multi-agent system that implements the BDI model for predicting bicycle trips, authors Ramón A. Briseño, Juan C. López, Rocio Maciel Arellano, Víctor M. Larios, Raul J. Beltrán & J. Antonio Orizaga T., 18) Visual explanation of the prediction of an image classifier, authors Tonantzin Marcyda Guerrero Velázquez & Juan Humberto Sossa Azuela, 19) Autonomous reconstruction of a metric map of a competition track using artificial vision, authors Esther Guadalupe Diaz Sarmientos, Juan Humberto Sossa Azuela & Alberto Petrilli Barceló, 20) Design and development of a remote monitoring system implementing Internet of Things, authors Yehoshua Agilar Molina, Sergio Sosa & Josue Ernesto Galindo Morales, 21) Time2Vec Embedding on a Seq2Seq Recurrent Network for Pedestrian Trajectory Prediction, autore Victor Peñaloza, 22) Methodology to Control a Mobile Robot with Fuzzy Logic, authors Ana Claudia Zenteno Vázquez, Gustavo Trinidad Rubín Linares, María del Carmen Santiago Díaz, Judith Pérez Marcial, Yeiny Romero Hernández & David Luna Ramos, 23) Handcraft and automatic approaches for the recognition of leukemia images, authors Rocio Ochoa Montiel, Leonardo Martínez Ibarra, Humberto Sossa Azuela & Gustavo Olague, 24) Design of an artificial organ with application to nursing in the UAEM valle de chalco university center, authors Maribel Hernandez Arroyo & Marco Alberto Mendoza Pérez, 25) Optimization of the Coefficients of the Predictive Model of the Number of Daily Cases of Coronavirus Covid-19 in Mexico, authors Ana Claudia Zenteno Vázquez, Gustavo Trinidad Rubín Linares, María del Carmen Santiago Díaz, Judith Pérez Marcial, Yeiny Romero Hernández & Antonio Eduardo Álvarez Nuñez, 26) IoT Management Platform using Industria 4.0 techniques for precision agriculture, authors Ali Perez Gomez, Adriana Carolina Sanchez Cahuich & Janeth Jimenez Gomez, 27) IoT system for monitoring and analysis of light intensity in indoor plants, authors Gabriel Elías Chanchí Golondrino, Manuel Alejandro Ospina Alarcón & Wilmar Yesid Campo Muñoz.

Among the countries participating in this special issue are: Mexico, the Netherlands, Japan, Guatemala, Colombia and Cuba, as well as 15 national and international universities such as: University of Groningen, TECNM, UVM, University of Tokyo, UNIVA, IPN, UPQ, UNAM, UV, UAEM, BUAP, UACJ, UDG, UTM, CICESE.

Special thanks are extended to the Computer Research Center of the National Polytechnic Institute, as well as the technical committee and reviewer for their support.

Juan Humberto Sossa Azuela
Maria Elena González Bañales
Edgar Gonzalo Cossio Franco
Guest Editors

November 2020

A Data Fusion Approach of Physical Variables Measured through a Wireless Sensor Network

Thomas Alejandro Arias Pelayo, Guillermo Molero Castillo,
Everardo Bárcenas, Rocío Aldeco Pérez

Universidad Nacional Autónoma de México,
Facultad de Ingeniería,
Mexico

thomas.arias95@gmail.com,
{gmolero, raldeco}@fib.unam.mx,
ebarcenas@unam.mx

Abstract. At present, current data fusion methods are a useful tool for integrating data sources, prior to data analytics, and provide a unified view of an observed phenomenon or event. This paper presents the development of the Monte Carlo method, as a data fusion mechanism, obtained from a wireless sensor network. This network of sensors was designed and installed in a closed environment of human occupation. The data collected was of physical variables, such as temperature, humidity, and dust density, which were stored in the cloud through ThingSpeak, which is an open-source platform for the Internet of Things. As a result, it was succeeded in data fused properly and the method was evaluated through the root-mean-square error. Undoubtedly, fused values can be useful, for example, for the analysis of the thermal comfort of users in closed environments, where there are minimal ventilation rates and adequate indoor air quality is needed.

Keywords: Data Fusion, Monte Carlo Method, Wireless Sensor Network, Physical Variables.

1 Introduction

Today, the automation of industrial processes, the improvement of workforce capabilities and the development of new products through Artificial Intelligence benefits society as part of the digital transformation process in this era of the fourth industrial Revolution. However, of all the technologies to consider for their potential, without a doubt, wireless sensor networks (WSN) are proving their usefulness for measuring and storing data on a specific environment, as a key technology to implement the Internet of Things (IoT), whose main characteristic is its low energy consumption and its deployment in inaccessible locations or even integrated within structures.

It is important to note that currently, wireless sensor networks are autonomous devices that work collaboratively, distributed throughout an area of interest and

Thomas Alejandro Arias Pelayo, Guillermo Molero Castillo, Everardo Bárcenas, Rocío Aldeco Pérez

whose objective is to monitor physical or environmental parameters, such as temperature, sound, vibrations, pressure, movement, or pollutants. Each element of the sensor network communicates wirelessly, offering a flexible system, easy to install, and scalable in large quantities. However, as the volume of data grows, so does the need to combine data from sensors to extract useful information, allowing an understanding of the environment and making timely decisions.

Faced with this situation, data fusion systems are an efficient solution to this problem. Data fusion is the process of associating and combining data at various levels, which come from different sources, such [1]: sensors, databases, signals, decisions, and others. This field of knowledge is used in different areas, such as [2]: signal processing, statistics, inference, and artificial intelligence. It has greater advances in applications of automatic target recognition, autonomous vehicle navigation, remote sensing, virtual environments, smart homes, threat identification, among others.

The idea of combining data from various sources, such as a network of sensors, is to obtain a global and unified vision of the observed phenomenon. Therefore, data fusion provides a formalization to data integration, in such a way that the results help in the decision-making process about a certain event, object, or action [3].

This paper aims to provide a vision of the progress of the development of a method for data fusion of physical variables, obtained in a closed space of human occupation, such as a room of a house, office and other places of rest or work, through a wireless sensor network. In particular, the analysis of the three variables was carried out, such as temperature, humidity, and dust density.

The document is organized in the following way, Section 2 presents the background of data fusion, some of the most significant contributions of data fusion are mentioned, its applications are discussed, the use of wireless sensor networks and the related jobs. Section 3 describes the method established as a proposed solution. Section 4 presents the results obtained, based on an application example, and Section 5 summarizes some conclusions and future work.

2 Background

As mentioned, data fusion is about the integration of data that comes from different sources. The purpose is to have a unified and refined view of them, in order to analyze them and give a conclusion. The data fusion as a field of knowledge emerged with military and robotics applications [10].

At present, one of the areas that have given the greatest impetus to data fusion is data science, where methods are used to extract, transform and load data sources as part of the data engineering process, prior to data analytics [11, 12].

Another area, which has also benefited from data fusion, is IoT since this is a technology that depends on interconnected objects, capable of communicating for the collection of data about a certain event [13].

2.1 Wireless Sensor Network

A wireless sensor network is made up of a set of interconnected sensor nodes that cooperate to measure and control a certain physical context that surrounds them, thus allowing interaction between people and devices or machines. These sensors work within a field of action to capture and transform signals into data, which are sent through other nodes to a common output, such as the Internet or databases [14].

A sensor network uses wireless or wired communication, and can be physically organized within an area where events occur in different ways, such as [7]: a) distributed, where there is no central node, but if a node stops function, it does not affect any other; b) centralized, where all nodes are peripheral except the central node, which is the only one that emits information; c) decentralized, where there is a collective hub of nodes, if one node goes down, several nodes may stop working, but the entire network does not go down.

In recent years, along with the growth of the Internet of Things, wireless sensor networks are gaining relevance in the industry, academia, and society in general, as a global infrastructure that allows offering advanced services through the interconnection of physical and virtual objects, thanks to the interoperability of information technologies [15], which, in turn, guarantee security and privacy requirements [17]. Thus, the applications are varied, among which agriculture, smart cities, smart homes, smart environments, industry, logistics, environment, military solutions, security, health, among others, stand out.

2.2 Data Fusion

Data fusion is a field of knowledge used to combine data produced by one or more sensors or data sources so that you can have a better estimate of the amount being measured [16]. Current ideas of data fusion are an efficient solution alternative to the increasing availability of data volumes. This is to combine data from various sources to obtain a complete description of an observed phenomenon [7].

On the other side, various definitions have emerged in data fusion. One of these points out that it is a multi-level, multi-stage process for the detection, association, correlation, estimation, and combination of data from one or more sources to achieve complete estimates and evaluations [18]. Recent examples of data fusion cover a wide range of areas, such as engineering, medicine, traffic control, environment, artificial intelligence, robotics, among others.

There are, therefore, application areas that until recently have been little explored by the data fusion community, some of these are [20]: the development of systems capable of incorporating hard and soft data, systems of data fusion in dynamic environments and real-time processing, as well as the incorporation of data fusion in smart sensors.

2.3 Monte Carlo Method

The Monte Carlo method is a non-deterministic technique, used to approximate numerical estimates to a great variety of mathematical problems. This method is

Thomas Alejandro Arias Pelayo, Guillermo Molero Castillo, Everardo Bárcenas, Rocío Aldeco Pérez

named in reference to the Monte Carlo Casino (Monaco), for being the capital of the game of chance, as roulette is a simple generator of random numbers [19]. This method bases its operation on simulation, which consists of repeating the characteristics and behaviors of a real system. The objective is to imitate the behavior of real variables to, as far as possible, analyze or predict simple or complex future values.

Currently, this method is a fundamental part of the Ray Tracing algorithms for the generation of 3D images [21]. The method may vary depending on the application, but in general, it follows the following steps: i) it defines the domain of the inputs; ii) it entries with random values are generated, following a probability distribution that is within the domain; iii) the inputs in the function are evaluated, and iv) the calculated values are gathered.

Being a probabilistic analysis algorithm, the possibility of error is given by the number of errors divided by the number of times the test is performed. Following the central limit theorem, Monte Carlo has an absolute error given by:

$$\frac{1}{\sqrt{N}}.$$

This indicates that if the number of trials is increased, the probability of the error tends to zero [19]. An advantage of the Monte Carlo simulation is that it can give a good approximation of the exact value. In this way, this method can be used to measure reliability in a sensor network [8]. The equation used is the following:

$$(b - a) \frac{1}{N} \sum_{n=1}^{\infty} f(x_i),$$

where:

- $(b - a)$ are the intervals of the function,
- N is the number of tests,
- x_i is the random value that is evaluated in the function.

2.4 Related Works

A growing problem in data fusion is dealing with data inconsistency. For this, there are various methods that can be used, such as sending the same data packet a second time. Once sent, the similarities of the two packets are compared and outliers are removed. Another option is to compare previous data measurements and then give a certain level of confidence to the sensors that were used [4]. Another method is to assign a series of weights hierarchically to the sensors [5], but none of these methods deal with the conflict, from the point of view of inconsistency, generated by using different sensors. Table 1 summarizes some identified works, as part of the literature review.

Therefore, due to the promising future of wireless sensor networks, there is a natural interest to embrace data fusion applications with this type of technology to monitor different phenomena and events efficiently.

Table 1. Related Works.

Author	Description	Method used	Limitations
Motro <i>et al.</i> (2006) [6]	It is a data source integration system, called Fusionplex.	The process consists of finding inconsistencies in data tuples, that is, inconsistencies are detected in the instances and eliminated according to the criteria.	High computational cost to resolve conflicts in commercial applications.
Kumar <i>et al.</i> (2009) [4]	Detects data inconsistency in sensors before it is sent.	Uses a neural network to detect the inconsistency of data through its variance.	It does not analyze the inconsistency of the data.
Frikha <i>et al.</i> (2015) [5]	Assigns a hierarchy to sensors to eliminate subjectivity and reduce inconsistency. The higher the hierarchy, the greater the weight.	The sensors were compared in pairs using matrices. The weights of the sensors are added.	It is required to define the criteria to assign the initial hierarchies.
Bakr <i>et al.</i> (2017) [7]	Three data fusion approaches are presented to deal with inconsistency: a) model-based, b) redundant, and c) fusion-based.	The treatment of inconsistency of data from various sources is described.	Lacks a process of fusing data from multiple sources.

Table 2. Characteristics of the NodeMcu board.

Characteristic	Value
1 Power supply	5 V
2 Output voltage on pins	3.3 V
3 Processing speed	80 - 160 MHz
4 Size	29.1 x 63.3 mm

3 Method

The working method defined for data fusion of physical variables, measured in a closed environment of human occupation, was divided into four stages: a) installation of the sensor network, b) data collection, c) data fusion, and d) evaluation of the fusion method.

a) Installation of the sensor network

In this stage, two sensor nodes were installed at the ends of a room. The NodeMcu board (ESP8266) was programmed to send data to the ThingSpeak platform, which allows storing, visualizing, and analyzing data in real-time, using cloud services for IoT applications. Using the cloud has advantages, since it allows access to the data collected from anywhere in the world, using any device with an Internet connection. In addition, it provides useful tools for users. Table 2 summarizes the main characteristics of the NodeMcu board, the required values of voltage, speed, and size. This board is used for the development of mobile IoT applications.

Table 3. Characteristics of the DHT11 sensor.

	Characteristic	Value
1	Power supply	3.3 – 5 V
2	Output voltage on pins	3.3 V
3	Humidity range	20 – 90% with 5% precision
4	Temperature range	0 – 50°C with $\pm 2^\circ\text{C}$ of precision
5	Size	14 x 18 x 5.5 mm

Table 4. Characteristics of the dust sensor.

	Characteristic	Value
1	Power supply	5 V
2	Current consumption	Icc: MAX 20 mA
2	Pulse cycle	10 \pm 1ms
3	Size	46.0 x 30.0 x 17.6 mm

Table 5. Extract of data collected through the implemented sensor network.

Date	Temperature (°C)	Humidity (%)	Dust Density (mg/m ³)
2020-05-30 19:54:38	28.9	67	0.02
2020-05-30 19:59:38	29.1	69	0.01
2020-05-30 20:04:39	29.0	71	0.02
...
2020-06-23 09:37:12	32.1	64	0.01
2020-06-23 09:42:13	32.2	65	0.01
2020-06-23 00:47:13	32.2	66	0.02

The DHT11 sensor includes a humidity and temperature measurement component that connects to an 8-bit microcontroller to ensure quality and fast response using low voltage. Table 3 summarizes the main characteristics of the DHT11 sensor, it indicates the power values required to make it work, its size, and the range of humidity and temperature values that it can measure.

On the other side, the GP2Y1010AU0F sensor was used to detect reflected light from dust and smoke particles. This type of sensor can distinguish between dust and smoke particles by means of a pulse pattern in the output voltage. Table 4 presents some of its most important characteristics, such as the pulse cycle it uses to detect dust particles, size and feed.

b) Data collection

To send data to the ThingSpeak channels and maintain the integrity of the data, the platform has a function called “thingSpeakRead” that imports the non-null values. The data that comes from the two nodes were grouped. To carry out this task, two channels were created, where the data was stored and public access was granted for it to be read by any user [22, 23]. The data was collected in a period of five weeks, from

May to June 2020. Table 5 shows, as an example, an extract of the captured data.

c) Data Fusion

As part of the data fusion process, the Monte Carlo method was used to verify the variation in the data that was obtained through the sensor nodes. This method was chosen for its ability to numerically operate complex systems, maintaining the input and output relationship, with a certain degree of uncertainty. Subsequently, the data were fused for each of the collected variables, that is, temperature, humidity, and dust density.

The Monte Carlo method was chosen since it allows us to identify what is going to happen and its probability of occurrence. In addition, it is important to point out that through this method it was possible to identify possible conflicts generated in the data, since if this situation is ignored, the resulting data could represent, in an erroneous way, the real data obtained in the experimentation.

d) Evaluation of the fusion method

The Root Mean Square Error (RMSE) was used to evaluate the fusion method. Through this measurement, the difference between the estimated (predicted) values by the Monte Carlo method and the values captured by the sensors is calculated. The formula used was the following:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^n \left(\frac{d_i - f_i}{\sigma_i} \right)^2},$$

4 Results

The results of the data fusion for each of the variables analyzed are shown below.

a) Temperature

The probability distributions of the values obtained for temperature were plotted, as well as the data fusion. Fig. 1 shows the measurements of sensors 1 (blue) and 2 (green), which are close to the 50th percentile of each. While the values of the data fusion (red) are between the measurements of the two sensors, with a slight similarity to the values obtained from sensor 2.

b) Humidity

The probability distributions of the humidity percentages were plotted, as well as the data fusion (Fig. 2). The most probable values for sensor 1 (blue) and sensor 2 (green) were found to be within the values of the data fusion (red). Sensor 1 (blue) measurements have average behavior relative to sensor 2 and data fusion.

c) Dust Density

Fig. 3 shows the probability distribution of the dust density percentages measured by the two sensors, as well as the values of the data fusion. Sensor 2 (green) values were found to be unusual, as they are skewed to the right.

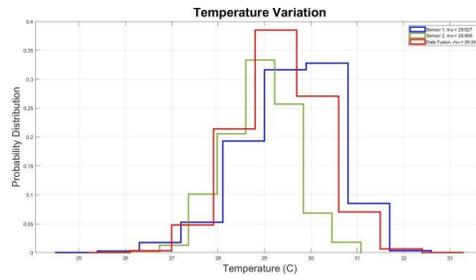


Fig. 1. Probability distribution of temperature data fusion.

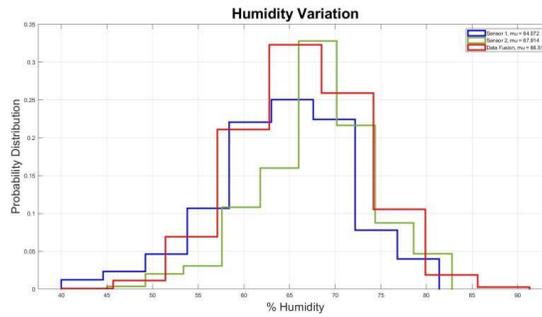


Fig. 2. Probability distribution of humidity data fusion.

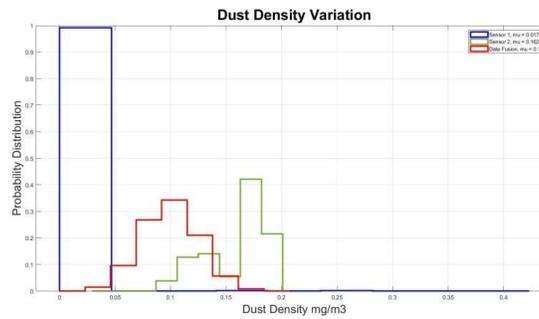


Fig. 3. Probability distribution of dust density data fusion.

Table 6. Mean square error estimation.

Variable	Sensor	RMSE
Temperature	Sensor 1	1.3892
	Sensor 2	1.2586
Humidity	Sensor 1	10.1487
	Sensor 2	9.2745
Dust density	Sensor 1	0.0895
	Sensor 2	0.0735

This may be due to sensor 2 being located above a bookcase and that the amount of dust that was raised was greater due to the cleaning carried out in this part of the room. In addition, it was observed that sensor 1 (blue) presented less variation, with a probability of density less than 0.05, that is, almost 99%. Although most of the data obtained in sensor 1 varied from 0.00 to 0.05, there were also some cases of variability between 0.07 to 0.4. This difference, with respect to the other sensor, is due to the fact that in that area of the room there is almost no air current and no cleaning was done.

On the other side, Table 6 shows the results obtained from the mean square error measurements with respect to the data fusion. Low levels of error were reached in temperature (1.38 and 1.25%) and dust density (0.08 and 0.07%); while, for humidity higher percentages of error were observed, that is, 9.27 and 10.14%. This may be because some days it rains and others don't. When it rains, the humidity increases, that is, there is high atmospheric saturation, which can give unusual values. Consequently, the precision level for temperature was 98.7%, humidity 90.3%, and dust density 99%. Given that 99% of the density levels of the powder were between 0.00 and 0.05, the RMSE value for the two sensors that measured this variable was low, that is, almost 1% error.

It is important to mention that the accuracy of the Monte Carlo method increases as more simulations are performed. Therefore, the computational cost also increases.

5 Conclusion

A network of sensors was implemented in a rest and work area, given the current context of the COVID-19 disease, such as a room in a home, in order to obtain data on temperature, humidity and dust density. For this, a NodeMcu development card (ESP 8266) with Wi-Fi connection capabilities was used in order to connect to the ThingSpeak platform to collect, visualize and analyze data from the sensors.

DHT11 sensors were used to measure temperature and humidity, and GP2Y1010AUOF sensors were used to measure dust density. The data obtained by the sensors can be consulted in real time through the two channels created in ThingSpeak. Sensor data can also be downloaded through the "Extract Data" function on the same channel

The values obtained for temperature, humidity, and density of the powder were correctly fused using the Monte Carlo method. The values of the sensors and the

Thomas Alejandro Arias Pelayo, Guillermo Molero Castillo, Everardo Bárcenas, Rocío Aldeco Pérez

values obtained through the method used were compared. In addition, low error levels were reached, which were evaluated by means of the root-mean-square error, which allowed to verify the accuracy of the obtained values. As future work, it will be sought to analyze based on the fused values of temperature, humidity, and dust density, if the observation environment, in this case, a room, is within the thermal comfort zone defined according to the ANSI/ASHRAE 55 standard. This type of analysis is useful to identify minimum ventilation rates and other measures to provide adequate indoor air quality in human-occupied spaces.

It will also consider expanding the implementation with different techniques, such as Artificial Neural Networks, Kalman Filter, among others, with the purpose of comparing the results and obtaining better and better-fused data.

Acknowledgment. This work was supported by UNAM-PAPIIT IA105320.

References

1. Kale, D., Aparadh, S.: a study of a detection and elimination of data inconsistency in data integration. *International Journal of Scientific Research in Science, Engineering and Technology*, 2(1), pp. 532-535 (2016)
2. Shahbazian, E., Bergeron, P., Duquet, J., Jouan, A., Valin, P.: Data fusion applications for military and civilian purposes developed DND/LM Canada decision support testbed. In: *Conference Record of the Thirty-Third Asilomar Conference on Signals, Systems, and Computers*, 1, pp. 420-424 (1999)
3. Durrant-Whyte, H., Henderson, T.: *Multisensor data fusion*. In: Siciliano B., Khatib O. (eds) *Springer Handbook of Robotics*. Springer, Berlin, Heidelberg (2008)
4. Kumar, M., Garg, D.: Multi-sensor data fusion in presence of uncertainty and inconsistency in data. In: Milisavljevic (Ed.), *Sensor and data fusion*, pp. 225-244, Intech (2009)
5. Frikha, A., Moalla, H.: Analytic hierarchy process for multi-sensor data fusion based on belief function theory. *European Journal of Operational Research*, 241(1), pp. 133-147 (2015)
6. Motro, A., Anokhin, P.: Fusionplex: resolution of data inconsistencies in the integration of heterogeneous information sources. *Information Fusion*, 7(2), pp. 176-196 (2006)
7. Bakr, M., Lee, S.: Distributed multisensor data fusion under unknown correlation and data inconsistency. *Sensors*, 17(11), pp. 1-31 (2017)
8. Yang, Q., Chen, Y.: Monte carlo methods for reliability evaluation of linear sensor systems. *IEEE Transactions on Reliability*, 60(1), pp. 305-314 (2011)
9. Schiavon, S., Hoyt, T., Piccioli, A.: Web application for thermal comfort visualization and calculating according to ASHRAE Standard 55. *Building Simulation*, 7(4), pp. 321-334 (2014)
10. Muñoz, J.: *Estimación Bayesiana para el tratamiento de la inconsistencia en la fusión de flujos de datos de fuentes físicas*. Doctoral Thesis, Universidad Veracruzana (2020)
11. Jagadish, H., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J., Ramakrishnan, R., Shahabi, C.: Big data and its technical challenges. *Communications of the ACM*, 57(7), pp. 739-749 (2014)
12. Zhu, X., Wu, G., Ding, W.: Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 26(1), pp. 97-107 (2014)
13. Cecchinell, C., Jimenez, M., Mosser, S., Riveill, M.: An architecture to support the collection of big data in the internet of things. In: *IEEE World Congress on Services*, pp. 442-449 (2014)

14. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. *Computer Networks*, 38(4), 393–422 (2002)
15. Ramos, A., Varum, T., Matos, J.: Compact Multilayer Yagi-Uda Based Antenna for IoT/5G Sensors. *Sensors*, 18(9), 2914 (2018)
16. Wah-Ng, G., Tan, C., Poh-Ng, T., Siow, S.: Assessment of Data Fusion Systems. In: 9th International Conference on Information Fusion, 1–8 (2006)
17. Contreras-Masse, R., Ochoa-Zezzatti, A., García, V., Elizondo, M.: Selection of IoT Platform with Multi-Criteria Analysis: Defining Criteria and Experts to Interview. *Research in Computer Science*, 148(11), 9–19 (2019)
18. Blasch, E., Steinberg, A., Das, S., Llinas, J., Chong, C., Kessler, O., Waltz, E., White, F.: Revisiting the JDL model for information exploitation. In: Proceedings of the 16th International Conference on Information Fusion, IEEE, 129–136 (2013)
19. Illana, J.: Metodos Monte Carlo. Departamento de Física Teórica y del Cosmos, Universidad de Granada, Spain, 1–47 (2013)
20. Barnaghi, P., Tonjes, R., Holler, J., Hauswirth, M., Sheth, A., Anantharam, P.: Real Time IoT Stream Processing and Large-scale Data Analytics for Smart City Applications. In: European Conference on Networks and Communications, 1–36 (2014)
21. Kajiya, J.: The Rendering Equation. *ACM SIGGRAPH Computer Graphics*, 20(4), 143–150 (1986)
22. Channel for sensor 1, https://thingspeak.com/channels/1010164/private_show. Last accessed in August 10, 2020
23. Channel for sensor 2, https://thingspeak.com/channels/1010156/private_show. Last accessed in August 10, 2020

Anexo B

En este apartado se presenta el código del método de Monte Carlo utilizado para la fusión de datos de las variables temperatura, humedad y densidad de polvo; programados en Matlab.

1. Temperatura

```
% Channel ID to read data from
readChannelID1 = 1010164; %Canal Fusión1
readChannelID2 = 1010156; %Canal Fusión2
% Field ID
Temp_FieldID1 = 1;
Temp_FieldID2 = 1;

% Channel Read API Key
% If your channel is private, then enter the read API
% Key between the '' below:
% Las Llaves de canal, son privadas, por eso esta escrito en ***
readAPIKey1 = '*****';
readAPIKey2 = '*****'; %

% Si se cambian los bins se cambian las probabilidades
nbins = 9;

% Obtención de las temperaturas
% Fecha de data obtenida de mayo 19 2020 a 23 junio 2020
tempS1 = thingSpeakRead(readChannelID1,'Fields',Temp_FieldID1,...
    'DateRange',[datetime(2020,5,2,12,21,10),datetime(2020,6,16,12,21,10)]);
tempS2 = thingSpeakRead(readChannelID2,'Fields',Temp_FieldID1 ,...
    'DateRange',[datetime(2020,5,2,12,21,10),datetime(2020,6,16,12,21,10)]);

% Se generan los histogramas
dis1 =
    histogram(tempS1,nbins,'Normalization','probability','DisplayStyle','stairs',
    'LineWidth',2.5,'EdgeColor','b');
hold on
dis2 =
    histogram(tempS2,nbins,'Normalization','probability','DisplayStyle','stairs',
    'LineWidth',2.5,'EdgeColor','#77AC30');

n = numel(tempS1);

% Algoritmo de fusión de datos
% Se calcula el promedio de desviación estándar
stdHistFusion = (std(tempS1) + std(tempS2)) / 2;
% Se calcula el promedio
mediaTempFusion = (median(tempS1) + median(tempS2)) /2;
% Se generan los valores aleatorios que usan las dos variables anteriores
random_data = stdHistFusion.*randn(n,1) + mediaTempFusion;
```

```

disFusion = histogram(random_data,nbins,
'Normalization','probability','DisplayStyle','stairs','LineWidth',2.5,'EdgeColor','r');
xlabel('Temperature (C)','FontSize',20);
ylabel('Probability Distribution','FontSize',20);
title('Temperature Variation','FontSize',25);

grid on;
% Put up legend.
legend1 = sprintf('Sensor 1, mu = %.3f', mean(tempS1));
legend2 = sprintf('Sensor 2, mu = %.3f', mean(tempS2));
legend3 = sprintf('Data Fusion, mu = %.3f', mean(random_data));
legend({legend1, legend2, legend3});

n

% Se calcula la Raíz del Error Cuadrático Medio para la Temperatura
RMSE_Temp_Sensor1 = sqrt(mean((tempS1 - random_data).^2));
RMSE_Temp_Sensor2 = sqrt(mean((tempS2 - random_data).^2));

RMSE_Temp_Sensor1
RMSE_Temp_Sensor2

% Exactitud del algoritmo
Exactitud_Temp = 100 - (RMSE_Temp_Sensor1 + RMSE_Temp_Sensor2 )/2;
Exactitud_Temp

% Cálculo de la varianza
VarTemp1 = sum((tempS1 - random_data).^2) / n;
VarTemp2 = sum((tempS2 - random_data).^2) /n;

VarTemp1
VarTemp2

```

2. Humedad

```
% Channel ID to read data from
readChannelID1 = 1010164; %Canal Fusión1
readChannelID2 = 1010156; %Canal Fusión2
% Field ID
HumFieldID1 = 2;
HumFieldID2 = 2;

% Channel Read API Key
% If your channel is private, then enter the read API
% Key between the '' below:
% Las Llaves de canal, son privadas, por eso esta escrito en ***
readAPIKey1 = '*****';
readAPIKey2 = '*****'; %

% Si se cambian los bins se cambian las probabilidades
nbins = 9;

% Obtención de las temperaturas
% Fecha de data obtenida de mayo 19 2020 a 23 junio 2020
humS1 = thingSpeakRead(readChannelID1, 'Fields', HumFieldID1, ...
    'DateRange', [datetime(2020,5,2,12,21,10), datetime(2020,6,16,12,21,10)]);
humS2 = thingSpeakRead(readChannelID2, 'Fields', HumFieldID2, ...
    'DateRange', [datetime(2020,5,2,12,21,10), datetime(2020,6,16,12,21,10)]);

% Se generan los histogramas
dis1 =
    histogram(humS1,nbins, 'Normalization', 'probability', 'DisplayStyle', 'stairs',
    'LineWidth', 2.5, 'EdgeColor', 'b');
hold on
dis2 =
    histogram(humS2,nbins, 'Normalization', 'probability', 'DisplayStyle', 'stairs',
    'LineWidth', 2.5, 'EdgeColor', '#77AC30');

n = numel(humS1);

% Algoritmo de fusión de datos
% Se calcula el promedio de desviación estándar
stdHistFusion = (std(humS1) + std(humS2)) / 2;
% Se calcula el promedio
mediaHumFusion = (median(humS1) + median(humS2)) / 2;
% Se generan los valores aleatorios que usan las dos variables anteriores
random_data = stdHistFusion.*randn(n,1) + mediaHumFusion;

disFusion = histogram(random_data,nbins,
    'Normalization', 'probability', 'DisplayStyle', 'stairs',
    'LineWidth', 2.5, 'EdgeColor', 'r');
xlabel('% Humidity', 'FontSize', 20);
ylabel('Probability Distribution', 'FontSize', 20);
title('Humidity Variation', 'FontSize', 25);

grid on;
% Put up legend.
legend1 = sprintf('Sensor 1, mu = %.3f', mean(humS1));
legend2 = sprintf('Sensor 2, mu = %.3f', mean(humS2));
```

```

legend3 = sprintf('Data Fusion, mu = %.3f', mean(random_data));
legend({legend1, legend2, legend3});

n

% Se calcula la Raíz del Error Cuadrático Medio
RMSE_Hum_Sensor1 = sqrt(mean((humS1 - random_data).^2));
RMSE_Hum_Sensor2 = sqrt(mean((humS2 - random_data).^2));

RMSE_Hum_Sensor1
RMSE_Hum_Sensor2

% Exactitud del algoritmo
Exactitud_Humedad = 100 - (RMSE_Hum_Sensor1 + RMSE_Hum_Sensor2)/2;
Exactitud_Humedad

```

3. Densidad de polvo

```
% Channel ID to read data from
readChannelID1 = 1010164; %Canal Fusión1
readChannelID2 = 1010156; %Canal Fusión2
% Field ID
Polvo_FieldID1 = 3;
Polvo_FieldID2 = 3;

% Channel Read API Key
% If your channel is private, then enter the read API
% Key between the '' below:
% Las Llaves de canal, son privadas, por eso esta escrito en ***
readAPIKey1 = '*****';
readAPIKey2 = '*****'; %

nbins = 9;%bins, si se cambian los bins se cambian las probabilidades

% Obtención de las densidades de polvo
% Fecha de data obtenida de mayo 19 2020 a 23 junio 2020
dens_polvo_S1 = thingSpeakRead(readChannelID1,'Fields',Polvo_FieldID1,...
    'DateRange',[datetime(2020,5,2,12,21,10),datetime(2020,6,16,12,21,10)]);
dens_polvo_S2 = thingSpeakRead(readChannelID2,'Fields',Polvo_FieldID2,...
    'DateRange',[datetime(2020,5,2,12,21,10),datetime(2020,6,16,12,21,10)]);

% Se generan los histogramas

dis1 =
    histogram(dens_polvo_S1,nbins,'Normalization','probability','DisplayStyle','s
    tairs','LineWidth',2.5,'EdgeColor','b');
hold on
dis2 =
    histogram(dens_polvo_S2,nbins,'Normalization','probability','DisplayStyle','s
    tairs','LineWidth',2.5,'EdgeColor','#77AC30');

n = numel(dens_polvo_S1);

% Algoritmo de fusión de datos
% Se calcula el promedio de desviación estándar
stdHistFusion = (std(dens_polvo_S1) + std(dens_polvo_S2)) / 2;
% Se calcula el promedio
mediaDensFusion = (median(dens_polvo_S1) + median(dens_polvo_S2)) / 2;
% Se generan los valores aleatorios que usan las dos variables anteriores
random_data = stdHistFusion.*randn(n,1) + mediaDensFusion;

disFusion = histogram(random_data,nbins,
    'Normalization','probability','DisplayStyle','stairs','LineWidth',2.5,'EdgeCo
    lor','r');

xlabel('Dust Density mg/m3','FontSize',20);
ylabel('Probability Distribution','FontSize',20);
title('Dust Density Variation','FontSize',25);

grid on;
% Put up legend.
legend1 = sprintf('Sensor 1, mu = %.3f', mean(dens_polvo_S1));
legend2 = sprintf('Sensor 2, mu = %.3f', mean(dens_polvo_S2));
legend3 = sprintf('Data Fusion, mu = %.3f', mean(random_data));
legend({legend1, legend2, legend3});
```

```

n

% Se calcula la Raíz del Error Cuadrático Medio
RMSE_Polvo_Sensor1 = sqrt(mean((dens_polvo_S1 - random_data).^2));
RMSE_Polvo_Sensor2 = sqrt(mean((dens_polvo_S2 - random_data).^2));

RMSE_Polvo_Sensor1
RMSE_Polvo_Sensor2

% Exactitud del algoritmo
Exactitud_Polvo = 100 - (RMSE_Polvo_Sensor1 + RMSE_Polvo_Sensor2)/2;
Exactitud_Polvo

% Se calcula la varianza. Se agregó esto para verificar si habia un error de
interpretación
VarDensPolvo1 = sum((dens_polvo_S1 - random_data).^2) / n;
VarDensPolvo2 = sum((dens_polvo_S2 - random_data).^2) /n;

VarDensPolvo1
VarDensPolvo2

```

Referencias bibliográficas

- Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4), 393–422.
- Bakr, M. y Lee, S. (2017). Distributed Multisensor Data Fusion under Unknown Correlation and Data Inconsistency. *Sensors*, 17(11), 1–31.
- Bernardos, A., Tarrío, P. y Casar, J. (2008). A Data Fusion Framework for Context-Aware Mobile Services. En *Multisensor fusion and integration for intelligent Systems*, 606–613.
- Bokareva, T., Hu, W., Kanhere, S., Ristic, B., Gordon, N., Bessell, T., Rutten, M., Jha, S. (2006). *Wireless Sensor Networks for Battlefield Surveillance*.
- Blasch, E., Steinberg, A., Das, S., Llinas, J., Chong, C., Kessler, O., Waltz, E., White, F. (2013). Revisiting the JDL model for information exploitation. In: *Proceedings of the 16th International Conference on Information Fusion*, 129–136. IEEE, Istanbul.
- Bogaert, P. y Gengler, S. (2017). Bayesian Maximum Entropy and Data Fusion for Processing Qualitative Data: Theory and Application for Crowdsourced Cropland Occurrences in Ethiopia. *Stochastic Environmental Research and Risk Assessment*, 1–17.
- Branton, C., Ullmer, B., Wiggins, A., Rogge, L., Setty, N., Beck, S. y Reeser, A. (2013). Toward Rapid and Iterative Development of Tangible, Collaborative, Distributed User Interfaces. *Symposium on Engineering Interactive Computing Systems*, 239–248.
- Cecchinell, C., Jimenez, M., Mosser, S. y Riveill, M. (2014). An Architecture to Support the Collection of Big Data in the Internet of Things. En *Services*, 442–449.
- Chen, M., Mao, S., y Liu, Y. (2014). Big Data: A Survey. *Mobile Networks and Applications*, 19(2), 171–209.
- Durrant-Whyte, H. y Henderson, T. (2001). *Multi Sensor Data Fusion*. Australian Centre for Field Robotics, The University of Sidney, Australia.
- Frikha, A., y Moalla, H. (2015). Analytic Hierarchy Process for Multi-Sensor Data Fusion Based on Belief Function Theory. *European Journal of Operational Research*, 241(1), 133–147.
- Gellersen, H., Schmidt, A. y Beigl, M. (2002). Multi-Sensor Context-Awareness in Mobile Devices and Smart Artifacts. *Mobile Networks and Applications*, 7(5), 341–351.
- Guerriero, M., Wheeler, F., Koste, G., Dekate, S., y Choudhury, N. (2016). Bayesian Data Fusion for Pipeline Leak Detection. *19th International Conference on Information Fusion*, 1(1), 278-285.
- Illana, J. (2013) *Métodos Monte Carlo*. Departamento de Física Teórica y del Cosmos, Universidad de Granada, España, 1–47.

- Jagadish, H., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J., Ramakrishnan, R., y Shahabi, C. (2014). Big Data and Its Technical Challenges. *Communications of the ACM*, 57(7), 86–94.
- Khattak, A., Akbar, N., Aazam, M., Ali, T., Khan, A. M., Jeon, S., ..., Lee, S. (2014). Context representation and fusion: Advancements and opportunities. *Sensors*, 14(6), 9628-9668.
- Khaleghi, B., Khamis, A., Karray, F. y Razavi, S. (2013). Multisensor Data Fusion: A Review of the State-of-the-Art. *Information Fusion*, 14(1), 28–44.
- Kumar, M., Garg, D. y Zachery, R. (2007). A Method for Judicious Fusion of Inconsistent Multiple Sensor Data. *Sensors Journal*, 7(5), 723–733.
- Kumar, M. y Garg, D. (2009). Multi-Sensor Data Fusion in Presence of Uncertainty and Inconsistency in Data. In: Milisavljevic (Ed.), *Sensor and data fusion*, 225–244.
- Lahat, D., Adali, T., y Jutten, C. (2015). Multimodal Data Fusion: An Overview of Methods, Challenges, and Prospects. *Proceedings of the IEEE*, 103(9), 1449–1477.
- Llinas, J., Hall, D. (1998). An introduction to multi-sensor data fusion. *Proceedings of the IEEE International Symposium on Circuits and Systems*, <https://doi.org/10.1109/iscas.1998.705329>
- Liggins, M., Hall, D., y Llinas, J. (2008). *Handbook of Multisensor Data Fusion: Theory and Practice*. CRC press.
- Motro, A., Anokhin, P. (2006). Fusionplex: resolution of data inconsistencies in the integration of heterogeneous information sources. *Information Fusion*, 7(2), 176–196.
- Muñoz-Benítez J., Molero-Castillo G., Benítez-Guerrero E., Bárcenas E. (2018). Data fusion as source for the generation of useful knowledge in context-aware systems. *Journal of Intelligent & Fuzzy Systems*, 34(5), 3165–3176.
- Rodríguez, S., De Paz, J., Villarrubia, G., Zato, C., Bajo, J., y Corchado, J. (2015). Multi-Agent Information Fusion System to Manage Data from a WSN in a Residential Home. *Information Fusion*, 23, 43–57.
- Salina, B. y Malathi, P. (2014). FPGA implementation of data fusion algorithm for object localization. *International Conference on Recent Advances and Innovations in Engineering*, <https://doi.org/10.1109/icraie.2014.6909200>
- Siciliano, B., y Khatib, O. (2016). *Handbook of Robotics*. Springer.
- Steinberg, A., Bowman, C., White, F. (1999). Revisions to the JDL data fusion model. *Sensor Fusion: Architectures, Algorithms, and Applications*, <https://doi.org/10.1117/12.341367>

- Surie, D., Pederson, T., Lagriffoul, F., Janlert, L., y Sjolie, D. (2007). Activity Recognition Using an Egocentric Perspective of Everyday Objects. En International conference on ubiquitous intelligence and computing, 246–257.
- Voida, S., Patterson, D., y Patel, S. (2014). Sensor Data Streams. En J. Olson y W. Kellogg (Eds.), Ways of Knowing. Springer Science & Business, 291–322.
- Wah-Ng, G., Tan, C., Poh-Ng, T., Siow, S. (2006). Assessment of Data Fusion Systems. 9th International Conference on Information Fusion, 1–8.
- Yang, M., Al-Kutubi, M., y Pham, D. (2013). Continuous Acoustic Source Tracking for Tangible Acoustic Interfaces. Measurement, 46(3), 1272– 1278.