





# Avances en Informática y Automática

Undécimo Workshop



# Avances en Informática y Automática

Undécimo Workshop

## Editores

Michael D. Galdámez Canales  
Diego Manuel Jiménez Bravo  
Lucía Martín Gómez  
Javier Pérez Marcos

Publicado en España por:

Departamento de Informática y Automática Facultad de Ciencias  
Universidad de Salamanca  
Plaza de los Caídos s/n  
37008, Salamanca, España  
Tel.: + 34 923294653  
Fax: + 34 923294514  
Web: <http://mastersi.usal.es>  
Email: [mastersi@usal.es](mailto:mastersi@usal.es)

**ISBN 978-84-09-02436-0**

Editores:

Michael D. Galdámez Canales  
Diego Manuel Jiménez Bravo  
Lucía Martín Gómez  
Javier Pérez Marcos

## Prólogo

El Máster Oficial en Sistemas Inteligentes de la Universidad de Salamanca tiene como principal objetivo promover la iniciación de los estudiantes en el ámbito de la investigación. El congreso organizado por el Departamento de Informática y Automática que se celebra dentro del Máster en Sistemas Inteligentes de la Universidad de Salamanca proporciona la oportunidad ideal para que sus estudiantes presenten los principales resultados de sus Trabajos de Fin de Máster y obtengan una realimentación del interés de los mismos.

La undécima edición del *workshop* “Avances en Informática y Automática”, correspondiente al curso 2016 - 2017, ha sido un encuentro interdisciplinar donde se han presentado trabajos pertenecientes a un amplio abanico de líneas de investigación, desde los sistemas biométricos y la visualización de la información hasta la minería de datos pasando por otros campos relacionados. Todos los trabajos han sido supervisados por investigadores de reconocido prestigio pertenecientes a la Universidad de Salamanca, proporcionando el marco idóneo para sentar las bases de una futura tesis doctoral. Entre los principales objetivos del congreso se encuentran:

- Ofrecer a los estudiantes un marco donde exponer sus primeros trabajos de investigación.
- Proporcionar a los participantes un foro donde discutir ideas y encontrar nuevas sugerencias de compañeros, investigadores y otros asistentes a la reunión.
- Permitir a cada estudiante una realimentación de los participantes sobre su trabajo y una orientación sobre las futuras direcciones de investigación.
- Contribuir al desarrollo del espíritu de colaboración en la investigación.

## Organización

El *workshop* “Avances en Informática y Automática” está organizado por el Departamento de Informática y Automática de la Universidad de Salamanca.

### Comité Organizador

Dr. José Luis Alonso Berrocal  
Dr. Francisco Javier Blanco Rodríguez  
Dr. Emilio S. Corchado Rodríguez  
Dra. María Belén Curto Diego  
Dr. José Rafael García-Bermejo Giner  
Dra. María Gracia Manzano Arjona  
Dr. Luis Antonio Miguel Quintales

## Índice general

Analítica Visual de Datos para Representación de la Interacción en una Red Social Privada y con Restricciones de Privacidad . . . . .	1
<i>Jorge Durán-Escudero, Francisco J. García-Peñalvo y Roberto Therón Sánchez</i>	
Integración de Modelos Difusos para la Evaluación de Aspectos Económicos en Controladores MPC No-Lineales . . . . .	14
<i>Lucía Fargallo Calvo, Pastora Isabel Vega Cruz y Silvana Roxani Rebollar</i>	
Sistema Conexionista Aplicado a la Identificación Biométrica . . . . .	29
<i>Michael Galdámez y Angélica González Arrieta</i>	
Sistema de Gestión de Consumo Adaptativo para la Gestión de la Demanda en Redes Eléctricas . . . . .	46
<i>Diego M. Jiménez-Bravo, Juan M. Corchado, Juan F. De Paz y Gabriel Villarrubia González</i>	
Sistema Inteligente para la Generación de Armonías . . . . .	57
<i>Lucía Martín-Gómez, Belén Pérez Lancho y María Navarro-Cáceres</i>	
Sistemas Inteligentes para la Resolución de Problemas en el Campo de la Biotecnología . . . . .	79
<i>Yeray Mezquita Martín y Emilio Santiago Corchado Rodríguez</i>	
Sistema de Recomendación de Música . . . . .	93
<i>Javier Pérez-Marcos, Vivian López Batista, y María N. Moreno García</i>	
Detección de Objetos para Coches Autónomos . . . . .	107
<i>Nicolás Ugrinovic Kehdy, William Raveane y Angélica González Arrieta</i>	
<b>Autores</b> . . . . .	<b>134</b>



# Analítica Visual de Datos para Representación de la Interacción en una Red Social Privada y con Restricciones de Privacidad

Jorge Durán-Escudero<sup>1</sup>, Francisco J. García-Peñalvo<sup>1</sup> y Roberto Therón Sánchez<sup>1</sup>

Departamento de Informática y Automática, Facultad de Ciencias  
Plaza de los Caídos s/n, 37008, Salamanca, España  
{jorge.d, fgarcia, theron}@usal.es

**Resumen** En este trabajo se realiza una propuesta para estudiar los datos que se van a generar en la red social privada y anónima del proyecto WYRED, con el fin de extraer conocimiento sobre cómo interaccionan sus usuarios, tanto entre ellos, como con la propia plataforma. Para ello se parte de la creación de un sistema que generará un conjunto de datos de prueba, lo más parecido posible al original. Con esta información y teniendo en cuenta el impacto de la privacidad a la hora de tratar los datos del proyecto, se ha propuesto una arquitectura flexible y completa para el desarrollo de las visualizaciones interactivas que van a permitir visualizar los datos anteriormente generados. Finalmente, se presentan varios casos de uso donde se demuestra la idoneidad de la analítica visual para realizar análisis de los datos del proyecto y extraer conocimiento, de manera sencilla.

**Keywords:** Analítica visual, redes sociales, arquitectura, generación de datos, interacción de usuarios, interacción en redes sociales.

## 1. Introducción

Hoy en día, las redes sociales son uno de los tipos de comunidades que mayor crecimiento están teniendo, gracias a la amplia difusión de las tecnologías de la información y la comunicación [22]. Sin embargo, las mismas siguen presentando algunos problemas, como la gestión de la privacidad o el análisis de los datos, para incrementar el conocimiento que se tiene de lo que está sucediendo dentro de ellas. Además, los expertos se encuentran con que, debido al volumen de información que generan, actualmente no es posible realizar análisis de manera manual de lo que ocurre en las mismas. Esto lleva a centrar este trabajo en la problemática de la gestión automática de estos datos y el planteamiento de un sistema que permita comunicarlos de manera efectiva.

El proyecto WYRED [8] consiste en el desarrollo de un ecosistema tecnológico, para poder conocer en mayor profundidad los intereses y problemas de los jóvenes, la manera que tienen de afrontarlos y, en definitiva, para ser un lugar donde su voz sea escuchada y tomada en cuenta.

Un ecosistema tecnológico es un conjunto de elementos tecnológicos que permiten cubrir todas las necesidades de un proyecto, para ello es necesario la gestión de los usuarios y la información generada, el soporte para la difusión de estos datos, la integración con otros ecosistemas tecnológicos y la capacidad de que cada uno de estos aspectos pueda evolucionar para adaptarse al proyecto [9]. En el caso de WYRED, este ecosistema está formado por 4 partes bien diferenciadas: un servicio que se encarga de anonimizar a los usuarios, una plataforma privada donde tienen lugar los diálogos con los jóvenes, un sistema para la difusión en redes sociales y una web pública para conocer el proyecto.

La propuesta de arquitectura que se presenta en este trabajo está centrada en la comunidad de WYRED, la cual es similar a un foro, ya que los usuarios organizan los debates en comunidades, hilos de discusión y comentario. Sin embargo, la comunidad también está preparada para albergar otras interacciones más complejas como diálogos sociales o proyectos de investigación. Al mismo tiempo, el proyecto tiene una serie de características que lo distinguen del resto [7], como su contexto internacional (varios idiomas, diferentes características socioculturales y muy variados puntos de vista) o la necesidad de salvaguardar la privacidad de los usuarios. Esto es necesario porque muchos de ellos serán menores y para transmitir que es un lugar donde pueden interactuar libremente. Debido a la gran cantidad de datos que el proyecto va a generar, se propone el uso de la analítica visual como una técnica efectiva para representar y extraer conocimiento [16].

El objetivo principal de este trabajo es plantear en estas primeras etapas del proyecto WYRED, una propuesta de la arquitectura de un sistema que permita dar soporte a la construcción de visualizaciones interactivas que ayuden a comprender mejor los datos, para anticiparse a las necesidades futuras del proyecto.

Esta arquitectura tiene que ser lo suficientemente flexible para poder adaptarse a las diversas características del proyecto, permitiendo además, construir sobre ella cualquier tipo de visualización que sea requerida, en este instante o en un futuro. Para ello debe ayudar a los investigadores en dos tareas principales:

- Conocer cómo evoluciona la comunidad y el contenido que se está generando.
- A ayudar en el proceso de la toma de decisiones.

Por ello, aunque el principal objeto de estudio del proyecto son los jóvenes, la arquitectura va a tener como usuarios finales, a los propios investigadores del proyecto. En línea con el objetivo final del proyecto, lo que se busca, en última instancia, es influir en las decisiones de los representantes públicos, para que tomen medidas que ayuden a mejorar la vida de los jóvenes, y en definitiva, que aprovechen sus aportaciones.

El artículo se organiza en las siguientes secciones: en primer lugar se trata la generación del conjunto de datos, después se presenta la propuesta de arquitectura, a continuación los resultados obtenidos y un caso de uso donde se demuestra el desempeño de la propuesta planteada y finalmente las principales conclusiones obtenidas y las futuras líneas de investigación.

## 2. Generación del Conjunto de Datos

Uno de los problemas que se ha tenido al realizar este trabajo, ha sido la no disposición de un conjunto de datos con el que realizar una propuesta de arquitectura. Es por ello que se ha tomado la decisión de intentar generarlo, para ello las principales maneras para afrontarlo son las siguientes:

- Utilizar los datos de una comunidad similar.
- Usar otras fuentes de datos que tengan características comunes.
- Generar los datos de manera artificial.

Extraer los datos de alguna comunidad similar es el proceso más simple y rápido para obtener un conjunto de datos. Para ello se pueden utilizar las principales redes sociales (Twitter, Facebook, etc.), las cuales han sido analizadas en profundidad por multitud de autores que han publicado sus conjuntos de datos [35] [19]. Pero esta solución no es válida en todos los casos, al tratarse de comunidades demasiado genéricas y de datos anonimizados.

Otros autores [34] han propuesto utilizar las entradas en los ficheros de registro, para generar el conjunto de datos. De manera que aquellas características que estén presentes en los registros y en el conjunto de datos objetivo, se mantengan y las que no aparezcan, sean generadas artificialmente. Este sistema posee la ventaja de que parte de los datos se corresponde con información real y, por tanto, es posible estudiarla para encontrar patrones y verificar hipótesis.

Otros investigadores centran sus estudios en generar el conjunto de datos por completo, de manera artificial. Dentro de este campo, hay que destacar a los que se centran en simular la interacción y los que además de lo anterior, intentan generar el contenido que se produciría. En el primer caso, han trabajado en modelar de forma matemática el crecimiento y la evolución de las interacciones en una red [30], lo que les permite alcanzar un conjunto de datos cuyo comportamiento es representativo. En el segundo caso, los autores se enfrentan a la alta complejidad que implica la generación de contenido, por ejemplo, de tipo textual, junto con la asignación de atributos representativos a cada individuo y sus interacciones. Aquí el principal exponente es LDBC-SNB Data Generator [29] el cual es un programa desarrollado para generar conjuntos de datos de comunidades para LDBC (*Linked Data Benchmark Council*) [5]. Para asignar a los atributos valores de manera lógica, los autores se basan en S3G2 [26] un *framework* que define la correlación que existe entre determinados atributos. Para la elección de los valores, el *software* cuenta con un conjunto de diccionarios donde están presentes los distintos valores que pueden tomar los atributos, seleccionando el valor final mediante diversas funciones que modelan la probabilidad de un suceso.

En un primer momento, para construir el conjunto de datos se intentó utilizar LDBC-SNB Data Generator, sin embargo, esto no fue factible, al generar un conjunto de datos que no es personalizable y que no contiene algunos de los atributos necesarios. Es por ello que se ha decidido construir desde cero el propio conjunto de datos, para ello se han dado los siguientes pasos:

1. Análisis de las entidades que hay que simular.

2. Identificación de sus principales atributos.
3. Creación del grafo de dependencia entre los mismos según el modelo descrito en S3G2 [26]. En la Fig. 1 se puede ver un ejemplo del mismo.
4. Asignación de valores a cada uno de los atributos según los atributos de los que dependen y los valores que estos presentan de manera usual. Para ello se han utilizado distintos indicadores como la población de un país, las expectativas de uso o distintos estudios sociológicos [18] [10].

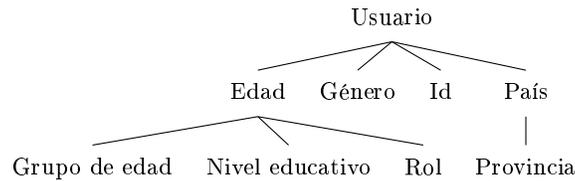


Figura 1: Dependencia entre los atributos de un usuario

### 3. Propuesta de Arquitectura

Una propuesta de arquitectura consiste en definir cada uno de los elementos de un sistema y cuál va a ser el modo en el que interaccionan los mismos. Este tipo de trabajo se vuelve necesario cuando se plantea la realización de un proyecto de cierto tamaño, ya que en él están presentes multitud de requisitos que se deben cumplir, para alcanzar un alto grado de satisfacción de los usuarios. En el caso de este proyecto, tiene que soportar un gran número de requisitos, siendo los principales:

- La capacidad de trabajar con distintas fuentes de datos.
- El soporte para gestionar la privacidad de los mismos.
- El análisis automático de los datos (en la medida de lo posible).
- La capacidad de representar los datos mediante visualizaciones interactivas.

Para soportarlos se ha decidido utilizar una arquitectura denominada de micronúcleo [32], que se basa en ofrecer una funcionalidad mínima en el núcleo, y complementar al mismo con un conjunto de componentes que son los que realizan las tareas requeridas por los usuarios. Este modelo presenta un cambio de filosofía respecto al patrón de capas, que se caracteriza por apilar las capas de manera horizontal, teniendo cada una de ellas un rol específico dentro de la aplicación.

La gran ventaja de aplicar esta arquitectura en este caso, es que el núcleo solo se va a encargar de obtener los datos y anonimizarlos, siendo cada uno de los componentes, los encargados de procesar esos datos y realizar la visualización correspondiente. Esto además permite conseguir una arquitectura muy flexible,

donde se pueda añadir fácilmente nuevas visualizaciones o eliminar alguna de las existentes, en el caso de que sus resultados no fueran satisfactorios [21].

Tomando como referencia la arquitectura de Docker (<https://goo.gl/LGk7vj>) se ha diseñado la propuesta reflejada en la Fig. 2, que consta de dos capas que forman el micrónúcleo y dos capas principales para cada uno de los componentes, que darán lugar a la generación de las visualizaciones interactivas.

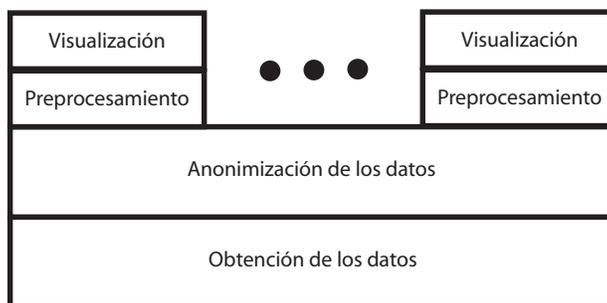


Figura 2: Esquema de la arquitectura propuesta

### 3.1. Obtención de los Datos

La obtención de los datos, en este proyecto implica algo más que unas simples consultas a una base de datos. Esto es debido a que la información del mismo se encuentra distribuida entre varios servicios, presentes en varias máquinas.

La información privada de los usuarios se almacena en un CAS, *Central Authentication Service* (<https://goo.gl/xD4Jkg>), siguiendo la estela de otros estudios que se han enfrentado a este problema [25] [12].

En el caso de la información pública de los usuarios, esta forma parte de la plataforma WYRED y está disponible en su base de datos. Finalmente, la información de la interacción de los usuarios con la plataforma, se almacena en una base de datos NoSQL, para afrontar de manera satisfactoria los problemas de escalabilidad [28] [3].

Esta capa del micrónúcleo, por tanto, tendrá que encargarse de fusionar los datos desde los distintos medios, además de la recuperación de la información.

### 3.2. Anonimización de los Datos

La capa encargada de anonimizar los datos es de vital importancia en este trabajo, al manejar datos que contienen información personal de los usuarios.

La manera de trabajar con parte de estos datos es sencilla, ya que cuestiones como el nombre, los apellidos o su correo electrónico, pueden ser eliminados sin perder información representativa. Sin embargo, esto no es suficiente para asegurar que los datos ya están anonimizados, ya que por la combinación de

los datos restantes puede ser posible identificar al usuario inicial [33]. Este tipo de datos que no son identificadores únicos, pero que tienen valores que no se suelen repetir (o su índice de repetición es bajo) en un conjunto de datos, se les denominan quasi-identificadores.

La propuesta para la anonimización de los datos consiste en analizar y detallar cuáles son los atributos quasi-identificadores que se van a tener, e intentar reducirlos:

- En el caso de la fecha de nacimiento, se propone transformar este dato en el año de nacimiento.
- En el caso del lugar de residencia, se plantea realizar un proceso similar, reduciendo la información a la provincia desde donde se participa.

Además de estas transformaciones, se propone que los resultados que se ofrezcan sean siempre  $k$ -anónimos con un valor de  $k=2$ . Lo que significa que no pueden existir registros con valores únicos, ya que, como mínimo, de cada registro deben existir 2 usuarios con iguales valores. La utilización de este valor permite asegurar la anonimidad de los datos, que serán publicados de manera abierta, para que otros investigadores puedan utilizarlos, tal y como estipula la Unión Europea para los proyectos financiados bajo el paraguas del Horizonte 2020 [24].

### 3.3. Módulo para el Análisis de los Temas Más Frecuentes

El análisis de los temas más frecuentes es una de las cuestiones más repetidas por los distintos investigadores. Algunos únicamente se centran en la evolución temporal de los mismos, sin embargo, otros investigadores consideran también muy importante la capacidad de poder explorar el uso de estos temas atendiendo a las características de los individuos (edad, género, país, etc.).

Gracias a la arquitectura propuesta anteriormente, este módulo es capaz de acceder a los datos de la plataforma para poder preprocesarlos. En este caso se plantea realizar un análisis automático de los temas más frecuentes usando LDA (*Latent Dirichlet Allocation*) [1]. Aunque esto plantea algunos problemas como la gestión de varios idiomas, en cuyo soporte han trabajado algunos autores [2][14] o la obtención de un nombre representativo para cada tema.

Para la propuesta de visualización, lo primero que se ha tenido en cuenta son sus principales tareas asociadas:

- Conocer la evolución de una temática: máximos, mínimos, patrones, etc.
- Poder comparar la evolución de varios temas.
- Ser capaz de medir la influencia de los atributos de los usuarios.

Teniendo en cuenta lo anterior, había que seleccionar un tipo de gráfico de entre los muchos existentes [17]. Por la importancia de la característica temporal, la primera decisión fue utilizar una representación que dispusiera de un eje horizontal donde poder mostrar cada uno de los instantes temporales. Pero todavía era necesario indicar cómo se iba a codificar la frecuencia de un tema, para lo cual había varias posibilidades como los gráficos de líneas, de áreas, o los histogramas.

En un principio se pensó utilizar una visualización basada en el concepto de *Theme River* [11]. Este sistema ya se había usado de manera efectiva en otros trabajos [4] [20], ya que permite identificar de manera sencilla los cambios de tendencia más importantes. Sin embargo, se ha demostrado que no es útil para detectar cambios de tendencia menores y, además, no permite representar un número amplio de temas. Para reducir estas desventajas, se ha combinado esta representación con otra basada en representar cada tema de manera individual, sobre líneas temporales paralelas [31]. Respecto a las capacidades de interacción y adaptación del gráfico, se proponen usar las siguientes:

- Capacidad de seleccionar los temas a representar.
- Posibilidad de realizar comparaciones según los atributos de los usuarios.
- Soporte para reordenar los temas, ya que es más sencillo comparar aquellos que están más próximos entre sí.
- Posibilidad de conocer el nivel de relevancia de ese tema en un instante.
- Capacidad de hacer *zoom* de manera automática.
- Posibilidad de restringir la selección a un periodo temporal.

### 3.4. Módulo para la Detección de Comunidades

Otro de los aspectos más relevantes a la hora de explorar una comunidad, es detectar las comunidades que implícitamente crean los usuarios. Para ello se han utilizado multitud de técnicas como el *clustering* jerárquico, la detección de nodos centrales o el cálculo de la centralidad [6], pero esto requiere de la ejecución de complejos algoritmos. Por ello se propone abordar esta tarea mediante la visualización interactiva.

La principal tarea que se quiere abordar con esta visualización es descubrir cómo interaccionan los usuarios, para poder intuir las comunidades implícitas que los mismos forman. Por esta razón, la representación mediante grafos se ha destacado como el mejor sistema para visualizar este tipo de datos. En el caso concreto de la visualización que se propone, las relaciones hacen referencia al número de comentarios que intercambian.

Respecto a la visualización se plantea codificar cada nodo con un tamaño relativo al número de mensajes que ha publicado en la plataforma. Además, la longitud de los enlaces debe representar la cercanía de un nodo respecto a otro, atendiendo a las interacciones que han tenido. Para ello se introduce el concepto de distancia relativa  $d_r$  entre dos nodos A y B, como un valor proporcional al número de enlaces totales de cada nodo entre el número de enlaces que comparten:

$$d_r(A, B) = k * \frac{g(A) + g(B)}{E(A, B)} \quad (1)$$

Respecto a las características de interacción se ha establecido las siguientes:

- Posibilidad de conocer las características de un usuario, al visitar su nodo.
- Capacidad de hacer *zoom* para analizar en mayor profundidad el grafo y ayudar a que esta visualización pueda escalar bien.

- Soporte para seleccionar un conjunto de nodos y explorarlos.
- Posibilidad de mover y analizar en detalle cada una de las comunidades que se formen.

### 3.5. Módulo para la Exploración de los Usuarios

El problema de representar los atributos de los usuarios de una plataforma es bastante complejo, debido al gran número de usuarios y características a mostrar. Por estas razones, se hace necesario el uso de una visualización que escale bajo demanda, compacta y sencilla a la hora de ser interpretada. Por ello se eligió utilizar las coordenadas paralelas [13], ya que permiten representar en un contexto bidimensional  $n$  dimensiones o atributos. Además se propone dotarlas de las siguientes características de interacción:

- Posibilidad de reordenar los atributos que van a ser visualizados, para así poder detectar si hay correlación entre ellos o no.
- Posibilidad de filtrar por cada uno de los atributos (filtrado múltiple).
- Capacidad de restringir el periodo temporal a estudiar.

### 3.6. Módulo para la Exploración Geográfica del Proyecto

Este módulo se encarga de dar respuesta a la necesidad de conocer cuáles son los países más activos y como afecta esta dimensión al análisis de los datos de la plataforma, siendo la visualización que mejor representa este concepto el mapa. Sin embargo, hay multitud de tipos de mapas, tanto atendiendo a las características que representan como a la proyección que utilizan. En la propuesta, se ha elegido utilizar la proyección Mercator, por ser la más familiar, para representar los países y las regiones del mundo. Además, se va a utilizar el color para representar el número de mensajes que han sido generados por los usuarios de cada uno de los territorios. Respecto a las características de interacción, se proponen las siguientes:

- Posibilidad de moverse por el mapa y volver al punto inicial.
- *Zoom* semántico, mostrando los países o las provincias según corresponda.
- Capacidad de conocer el número de mensajes exacto en cada país.
- Posibilidad de filtrar los datos por país.

## 4. Resultados

Para desarrollar la arquitectura propuesta, se ha recurrido a utilizar tecnologías y lenguajes de programación web. Esta decisión permite principalmente dos cosas: conseguir que los desarrollos sean accesibles a un mayor público y dotarlos de un mayor grado de interactividad.

Al realizar el desarrollo de cada uno de los módulos, hay un aspecto que ha tomado gran importancia, la capacidad de poder filtrar los datos. Para ello, se ha establecido en la parte superior unos controles destinados a tal fin, lo que ayuda

a cumplir con el mantra de la analítica visual enunciado por Keim *Analyze first, show the important, zoom, filter and analyze further, details on demand* [15].

La utilización de una arquitectura modular no implica, necesariamente, el uso de cada uno de los componentes por separado, por ello han sido combinados mediante la técnica de vistas enlazadas, para constituir un panel de monitorización que permita explorar todas las facetas del proyecto, al mismo tiempo, como se puede ver en la Fig. 3.

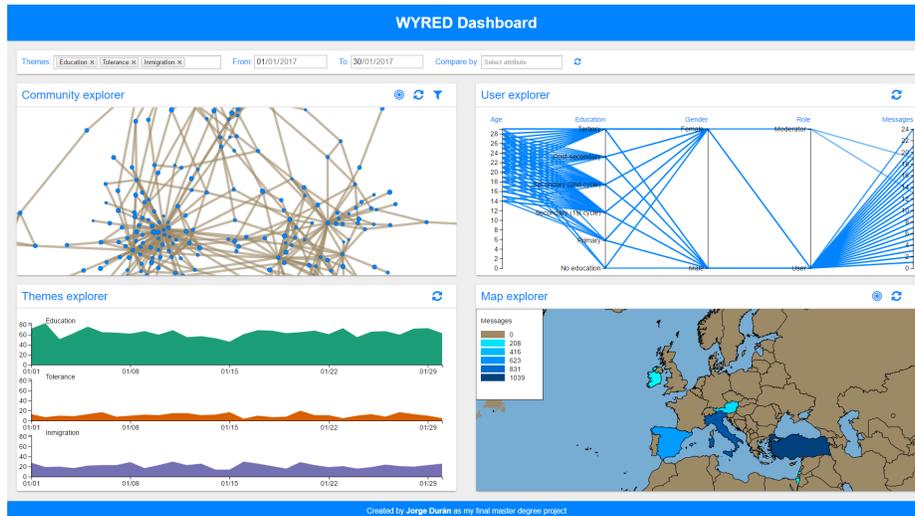


Figura 3: Panel de monitorización del proyecto<sup>1</sup>

Para demostrar cómo el sistema propuesto podría ser usado para resolver algunas preguntas de investigación típicas, a continuación se describe un caso de uso del sistema, cuya pregunta de investigación es *¿Cuáles son las principales comunidades sobre educación y empleo, y qué características tienen?*.

Lo primero de debe realizar un investigador es identificar los principales temas que están presentes en la pregunta de investigación, en este caso, educación y empleo. Por esta razón, tendrá que seleccionar ambos en el selector de temas, como se puede apreciar en el extremo izquierdo de la Fig. 4.



Figura 4: Selección de los temas para la pregunta de investigación

<sup>1</sup> Accesible en <https://jorge-duran.com/research/tfm/dashboard/>

Después, con la visualización creada por el explorador de comunidades podrá identificar las principales comunidades que se han formado, teniendo en cuenta los grupos de usuarios más compactos. En la Fig. 5a se puede apreciar que, fundamentalmente, los usuarios se agrupan en torno a 3 comunidades, las cuales han sido marcadas para facilitar la demostración. Para explorar una comunidad, el investigador deberá incluir en el rectángulo de selección, que aparece al hacer clic en la visualización, todos los puntos que, a su juicio, forman parte de esa comunidad, los cuales mantendrán su color azul para favorecer su identificación.

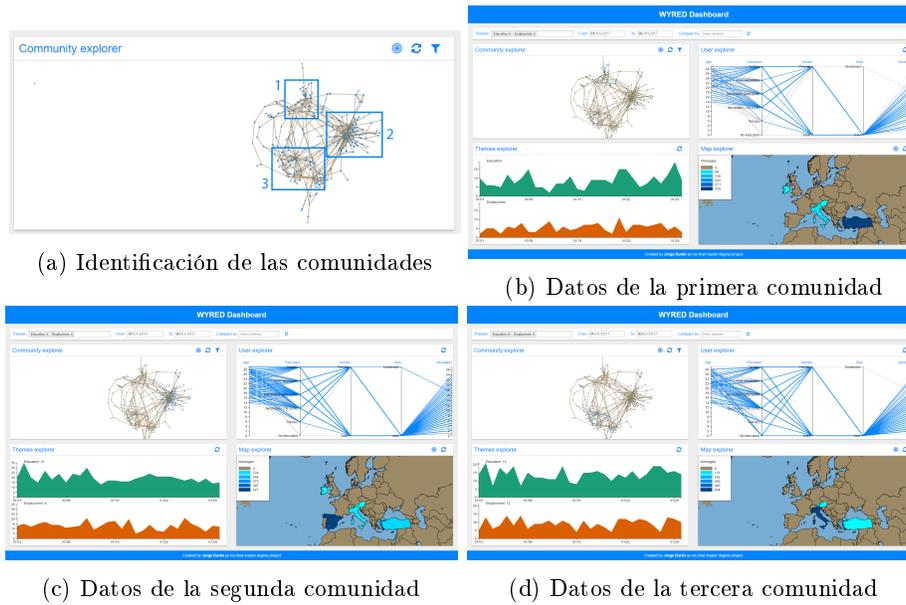


Figura 5: Principales comunidades y sus datos<sup>2</sup>

Al analizar los datos de la primera comunidad (Fig. 5b), se puede ver cómo los principales usuarios son de Turquía, debido a que este país es el que presenta un tono más oscuro en el mapa. Respecto a los temas, los mimos hablan más sobre educación que sobre empleo, como muestra el explorador de temas.

En el caso de la segunda comunidad (Fig. 5c), se puede ver como los usuarios son principalmente españoles, que en momentos puntuales hablan mucho sobre empleo, pero que de manera más continua, hablan más de educación, como se aprecia en el explorador de temas.

Si se observan los datos de la tercera comunidad (Fig. 5d), se puede apreciar como está constituida principalmente por italianos cuyo comportamiento es

<sup>2</sup> Accesible en <https://jorge-duran.com/research/tfm/Articulos/JorgeDuran/img/>

similar a los usuarios de la segunda comunidad. Sin embargo, la comunidad presenta un número inusualmente bajo de usuarios con educación postsecundaria, lo cual se puede ver al comprobar en el explorador de usuarios, que la mayor parte de las líneas que atraviesan la marca de postsecundaria están en gris.

Para mostrar las características interactivas de este desarrollo y como un investigador usaría estas visualizaciones para resolver la pregunta de este caso de uso, se ha grabado un vídeo (<https://goo.gl/js3hkp>) donde se puede comprobar todo el proceso.

## 5. Conclusiones y Futuras Líneas de Investigación

En este trabajo se ha presentado la propuesta de arquitectura para elaborar un conjunto de visualizaciones interactivas que permitan explorar los datos del proyecto WYRED. La cual se basa en la arquitectura de micronúcleo que consta de 2 capas básicas (adquisición y anonimización de datos) y de 4 módulos (exploración de temas, comunidades, usuarios y geográfica), que han permitido cumplir los objetivos planteados en un principio. Aunque el mismo podría ser ampliado con la puesta en marcha de un estudio de usabilidad con 5 usuarios [23], el soporte al trabajo colaborativo con las visualizaciones o permitiendo integrar el mismo con otros sistemas [27].

## 6. Agradecimientos

Este proyecto ha recibido fondos del programa de la Unión Europea de investigación e innovación Horizonte 2020 en virtud del acuerdo de subvención número 727066.

## Referencias

1. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.
2. J. Boyd-Graber and D. M. Blei, "Multilingual topic models for unaligned text," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2009, pp. 75–82.
3. R. Cattell, "Scalable sql and nosql data stores," *SIGMOD Rec.*, vol. 39, no. 4, pp. 12–27, May 2011.
4. W. Dou, X. Wang, R. Chang, and W. Ribarsky, "Paralleltopics: A probabilistic approach to exploring document collections," in *2nd IEEE Conference on Visual Analytics Science and Technology 2011, VAST 2011*, 2011, Conference Proceedings, pp. 231–240.
5. O. Erling, A. Averbuch, J. Larriba-Pey, H. Chafi, A. Gubichev, A. Prat, M.-D. Pham, and P. Boncz, "The ldbc social network benchmark: Interactive workload," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '15. New York, NY, USA: ACM, 2015, pp. 619–630.
6. S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3, pp. 75–174, 2010.

7. F. J. García-Peñalvo and J. Durán-Escudero, *Interaction Design Principles in WY-RED Platform*. Springer International Publishing, 2017, pp. 371–381.
8. F. J. García-Peñalvo and N. A. Kearney, “Networked youth research for empowerment in digital society. the wyred project,” in *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM’16)*. ACM, 2016, Conference Proceedings, pp. 3–9.
9. A. García-Holgado and F. J. García-Peñalvo, “The evolution of the technological ecosystems: An architectural proposal to enhancing learning processes,” in *Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality*, ser. TEEM ’13. New York, NY, USA: ACM, 2013, pp. 565–571. [Online]. Available: <http://doi.acm.org/10.1145/2536536.2536623>
10. S. Greenwood, A. Perrin, and M. Duggan. Social media update 2016. [Online]. Available: <http://www.pewinternet.org/2016/11/11/social-media-update-2016/>
11. S. Havre, E. Hetzler, P. Whitney, and L. Nowell, “Themeriver: visualizing thematic changes in large document collections,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 9–20, 2002.
12. F. Huang, C. x. Wang, and J. Long, “Design and implementation of single sign on system with cluster cas for public service platform of science and technology evaluation,” in *2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*, Nov 2011, pp. 732–737.
13. A. Inselberg and B. Dimsdale, *Parallel Coordinates for Visualizing Multi-Dimensional Geometry*. Tokyo: Springer Japan, 1987, pp. 25–44.
14. J. Jagarlamudi and H. Daumé III, “Extracting multilingual topics from unaligned comparable corpora,” in *European Conference on Information Retrieval*. Springer, 2010, pp. 444–456.
15. D. A. Keim, F. Mansmann, and J. Thomas, “Visual analytics: How much visualization and how much analytics?” *SIGKDD Explor. Newsl.*, vol. 11, no. 2, pp. 5–8, May 2010.
16. E. D. Keim, J. Kohlhammer, and G. Ellis, “Mastering the information age: Solving problems with visual analytics, eurographics association,” 2010.
17. K. Kucher and A. Kerren, “Text visualization techniques: Taxonomy, visual survey, and community insights,” in *2015 IEEE Pacific Visualization Symposium (PacificVis)*, 2015, Conference Proceedings, pp. 117–121.
18. A. Lenhart. Teens, social media & technology overview 2015. [Online]. Available: <http://www.pewinternet.org/2015/04/09/mobile-access-shifts-social-media-use-and-other-online-activities/>
19. J. Leskovec and A. Krevl, “SNAP Datasets: Stanford large network dataset collection,” <http://snap.stanford.edu/data>, Jun. 2014.
20. S. Liu, M. X. Zhou, S. Pan, W. Qian, W. Cai, and X. Lian, “Interactive, topic-based visual text summarization and analysis,” in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ser. CIKM ’09. New York, NY, USA: ACM, 2009, pp. 543–552.
21. K. Matkovic, W. Freiler, D. Gracanin, and H. Hauser, “Comvis: A coordinated multiple views system for prototyping new visualization technology,” in *12th International Conference Information Visualisation, IV08*, 2008, Conference Proceedings, pp. 215–220.
22. Micro Focus. How much data is created on the internet each day? [Online]. Available: <https://blog.microfocus.com/how-much-data-is-created-on-the-internet-each-day/>

23. J. Nielsen and T. K. Landauer, "A mathematical model of the finding of usability problems," in *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, ser. CHI '93. New York, NY, USA: ACM, 1993, pp. 206–213.
24. OpenAire. What is the open research data pilot? [Online]. Available: <https://www.openaire.eu/opendatapilot>
25. Z. A. Pardos and K. Kao, "Moocrp: An open-source analytics platform," in *2nd ACM Conference on Learning at Scale, L@S 2015*. Association for Computing Machinery, Inc, 2015, Conference Proceedings, pp. 103–110.
26. M.-D. Pham, P. Boncz, and O. Erling, "S3g2: A scalable structure-correlated social graph generator," in *Technology Conference on Performance Evaluation and Benchmarking*. Springer, 2012, Conference Proceedings, pp. 156–172.
27. W. A. Pike, J. Stasko, R. Chang, and T. A. O'connell, "The science of interaction," *Information Visualization*, vol. 8, no. 4, pp. 263–274, 2009.
28. J. Pokorny, "Nosql databases: a step to database scalability in web environment," *International Journal of Web Information Systems*, vol. 9, no. 1, pp. 69–82, 2013.
29. A. Prat and X. Sanchez. Ldbc-snb data generator. [Online]. Available: [https://github.com/ldbc/ldbc\\_snb\\_datagen](https://github.com/ldbc/ldbc_snb_datagen)
30. H. Pérez-Rosés and F. Sebé, "Synthetic generation of social network data with endorsements," *Journal of Simulation*, vol. 9, no. 4, pp. 279–286, 2015.
31. W. Ribarsky, D. Xiaoyu Wang, and W. Dou, "Social media analytics for competitive advantage," *Computers and Graphics (Pergamon)*, vol. 38, no. 1, pp. 328–331, 2014.
32. M. Richards, "Software architecture patterns," *O'Reilly Media*, 2015.
33. L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.
34. J. Yee, R. F. Mills, G. L. Peterson, and S. E. Bartczak, "Automatic generation of social network data from electronic-mail communications," DTIC Document, Report, 2005.
35. R. Zafarani and H. Liu. Social computing data repository at ASU. [Online]. Available: <http://socialcomputing.asu.edu>

# Integración de Modelos Difusos para la Evaluación de Aspectos Económicos en Controladores MPC No-Lineales

Lucía Fargallo Calvo<sup>1</sup>, Pastora Isabel Vega Cruz<sup>1</sup> y Silvana Roxani Rebollar<sup>1</sup>

Departamento de Informática y Automática, Facultad de Ciencias  
Plaza de los Caídos s/n, 37008, Salamanca, España  
{luciafargallo,pvega,mfs}@usal.es  
<http://diaweb.usal.es/>

**Resumen** En este trabajo se propone una técnica basada en lógica borrosa para la negociación entre los distintos controladores locales de un sistema de control distribuido, que utiliza controladores predictivos (MPC). Es una alternativa al juego cooperativo entre controladores donde cada MPC proporciona tres posibles soluciones para la variable manipulada, y mediante la negociación entre ellos se llega a una solución compromiso. La lógica borrosa se utiliza para introducir criterios económicos y restricciones del proceso en la negociación. Además, cuando el número de controladores es elevado, facilita la evaluación de los criterios económicos y de operación al interpretar información cualitativa. La validación de la técnica se realiza sobre un simulador de un sistema de cuatro tanques acoplados. Se evaluó la técnica propuesta frente a cambios de referencia y perturbaciones mostrándose un adecuado seguimiento de las referencias y reacción satisfactoria a perturbaciones.

**Keywords:** Control predictivo distribuido, juego cooperativo, sistema de cuatro tanques acoplados

## 1. Introducción

La industria demanda cada vez más mayor calidad y productividad, optimización de los costes, seguridad y respeto al medio ambiente. Los sistemas de control convencionales que utilizan un control PID solucionan muy bien la mayoría de los problemas de control básico, sin embargo, cuando nos encontramos con problemas más complejos en los que existe interacción se necesitan soluciones más elaboradas y por ello se requiere de sistemas de control avanzados. El control distribuido se presenta como una solución a los problemas complejos a gran escala, al estar formado por varios controladores que controlan individualmente un número de variables, los cuales tienen la capacidad de comunicarse entre ellos al encontrarse interconectados por una red de comunicación. Estos controladores son más seguros que los controladores centralizados, ya que estos últimos concentran todo el problema de control en único elemento (Kouvaritakis y Cannon, 2016 [4]). Aunque tradicionalmente se han utilizado controladores

PID distribuidos, actualmente se está incluyendo el control predictivo (MPC), ya que es una técnica que aporta eficiencia y flexibilidad y es aplicable además en sistemas multivariados y no lineales. El MPC incorpora un modelo interno del proceso para predecir el comportamiento futuro del proceso, y tiene la ventaja de que se pueden incorporar restricciones de entrada y/o salida en la formulación (Camacho, 2004 [2]).

Entre las múltiples técnicas de control predictivo distribuido está la de Maestre et al. (2011) [7], que plantea un juego cooperativo en el que dos controladores locales diferentes que se comunican entre sí para obtener la mejor solución para la variable manipulada. Los controladores locales son MPC lineales que proporcionan tres posibles soluciones de la variable manipulada para cada uno de los subsistemas, y mediante la comunicación entre ellos se intercambian esta información, con el fin de llegar a un acuerdo.

En este trabajo se propone una estrategia de control distribuido que basándose en Maestre et al. (2011) [7], se incluya un sistema de inferencia difuso en la negociación entre los controladores locales para seleccionar la mejor señal de control en cada periodo de muestreo. Cuando el número de controladores es pequeño, la evaluación del coste de cada una de las posibles combinaciones de variables manipuladas es asumible, pero para un número grande de controladores, el número de combinaciones hace que se deba realizar alguna modificación en el algoritmo. Además, el sistema difuso permite la incorporación de criterios económicos u otros que son difíciles de introducir en el algoritmo del juego cooperativo. Hay que indicar también que en este trabajo se ha estudiado el efecto de las perturbaciones, tanto internas como externas, sobre el caso de estudio elegido.

Para validar la metodología se ha considerado el problema de control de nivel de agua en los dos tanques inferiores de una planta formada por cuatro tanques acoplados (Johansson, 2000 [5]). A pesar de su simplicidad es un sistema muy interesante ya que es un sistema altamente acoplado y con dinámicas complicadas que hacen difícil su control. Se han propuesto diferentes soluciones para este problema de control de cuatro tanques, como son el control del modelo interno [3], el control robusto multi-variable [9], el control dinámico de matriz [8], control distribuido MPC [8] control centralizado y descentralizado [6], un esquema de sensibilidad impulsado por MPC distribuido [10] y un control MPC distribuido basado en el juego cooperativo [7]. En este caso, se evaluó la técnica propuesta frente a cambios de referencia y perturbaciones.

El sistema de inferencia difuso utilizado en el algoritmo propuesto tiene en cuenta la minimización de los costes de bombeo, el cumplimiento de un tiempo de residencia del agua adecuado y la minimización del riesgo de reboso en los tanques superiores como criterios para definir las reglas borrosas.

La estructura del artículo es la siguiente: en la sección 2 se presenta el marco teórico, el problema de control distribuido y el algoritmo de juego cooperativo, en la sección 3 se describe la planta y el problema de control a resolver, en la sección 4 se presenta la metodología propuesta, el desarrollo de la estrategia de negociación difusa. Los resultados se presentan en la sección 5 y las conclusiones en la sección 6.

## 2. Marco teórico

### 2.1. Control distribuido

El Control Predictivo (MPC) es un algoritmo de control que calcula una secuencia temporal discreta de las variables manipuladas sobre un horizonte futuro con el fin de optimizar el comportamiento de un sistema o proceso que se desea controlar. Utilizando un modelo del proceso para predecir las salidas del mismo en un horizonte de predicción  $N$ , calcula la secuencia de señales de control mediante la optimización de una cierta función objetivo. Se trata de un algoritmo de horizonte deslizante puesto que el proceso de cálculo se repite en cada instante a medida que se toman nuevas medidas de las variables a controlar. Además, permite tener en cuenta restricciones de operación del proceso.

La función de coste o función objetivo es una forma cuadrática que tiene en cuenta tanto los errores de seguimiento en los incrementos de nivel, como el esfuerzo de control, lo cual depende de las entradas a aplicar en un horizonte. Suponiendo que los horizontes de predicción y control coinciden:

$$J = \sum_{k=0}^{N-1} [(x_{refk} - x_k)^T Q (x_{refk} - x_k) + (u_{refk} - u_k)^T R (u_{refk} - u_k)] + x_N^T P x_N \quad (1)$$

Donde  $x_{ref}$  es el estado de referencia y  $u_{ref}$  son las referencias para la variable manipulada.  $Q$  y  $R$  sirven para penalizar más o menos los errores de seguimiento y los esfuerzos de control (entradas) y se utilizarán como parámetros de ajuste del controlador, del mismo modo que el horizonte de control  $N$ . La matriz  $P$  se añade como coste terminal para asegurar la estabilidad del controlador

El problema de optimización calcula la secuencia de entradas  $U$  que minimice la función de coste (1) sujeta a restricciones sobre las entradas y los estados. Este es un problema de programación cuadrática o  $QP$ , en el que la función de coste se puede poner en función de  $U$ .

### 2.2. Planteamiento del MPC distribuido

En el problema de control distribuido para dos subsistemas acoplados por las entradas, cada controlador solo tiene información parcial del modelo y de los estados del sistema, por lo que ambos se deben comunicar para encontrar una solución al problema de optimización que proporciona la señal de control. En general, en estos problemas cada controlador resuelve en cada tiempo de muestreo un problema de optimización asumiendo una entrada fija para el otro subsistema.

**Los modelos de cada subsistema son:**

$$x_1(t+1) = A_1 x_1(t) + B_{11} u_1(t) + B_{12} u_2(t) \quad (2)$$

$$x_2(t+1) = A_2x_2(t) + B_{22}u_2(t) + B_{21}u_1(t) \quad (3)$$

Donde  $x_i \in R_{n_i}$ , son los estados de cada subsistema y  $u_i \in R_{n_i}$  son las diferentes entradas. Las variables e índices de interés en el problema de control distribuido son [7]:

$U_i$  : variables de decisión en la optimización resuelta por los dos controladores.

$$U_1 = \begin{matrix} u_{1,0} \\ u_{1,1} \\ \cdot \\ \cdot \\ u_{1,N-1} \end{matrix} \quad (4)$$

$$U_2 = \begin{matrix} u_{2,0} \\ u_{2,1} \\ \cdot \\ \cdot \\ u_{2,N-1} \end{matrix} \quad (5)$$

$n_i$  : controlador vecino del controlador  $i$ ;

es decir  $U_{n_1} = U_2$  y  $U_{n_2} = U_1$

$J$  : función de coste local del controlador  $i$ :

$$J_1(x_1), U_1, U_2 = \sum_{k=0}^{N-1} L_1(x_{1,k}, u_{1,k}) + F_1(x_{1,N}) \quad (6)$$

$$J_2(x_2), U_2, U_1 = \sum_{k=0}^{N-1} L_2(x_{2,k}, u_{2,k}) + F_2(x_{2,N}) \quad (7)$$

Donde  $L_i$  y  $F_i$  con  $i = 1, 2$ , se definen como:

$$L_i(x, u) = x^T Q_i x + u^T R_i u \quad (8)$$

$$F_i(x) = x^T P_x \quad (9)$$

Con  $Q_i, P_i > 0, R_i > 0$ . El horizonte de predicción es  $N$ .

A continuación, se describe el algoritmo propuesto por J.M. Maestre et al. (2011) [7] basado en el juego cooperativo para resolver el problema de control. Se parte de este algoritmo a la hora de implementar la coordinación basada en lógica difusa. Se define en primer lugar las distintas secuencias de entrada:

$U_i^d(t)$ : Secuencia de entrada óptima del controlador  $i$  en el tiempo  $t$ .

$U_i^s(t)$ : Secuencia óptima de entrada del controlador  $i$  desplazada, obtenida a partir de la secuencia óptima de entrada del controlador  $i$  en el tiempo  $t-1$  denotada como  $U_i^d(t-1)$ . Detalladamente, el algoritmo es el siguiente:

1. En cada instante  $t$ , cada controlador  $i$  recibe su correspondiente medida del estado parcial  $x_i(t)$ .
2. Cada controlador  $i$  minimiza  $J_i$  suponiendo que el controlador vecino sigue aplicando la trayectoria óptima del paso anterior. El controlador 1 resuelve el siguiente problema de optimización:

$$\begin{aligned}
 U_1^*(t) &= \operatorname{argmin}_{U_1} J_1(x_1(t), U_1, U_2^s(t)) \\
 x_{1,k+1} &= A_1 x_{1,k} + B_{11} u_{1,k} + B_{12} u_{2,k} \\
 x_{1,0} &= x_1(t) \\
 x_{1,k} &\in X_1, k = 0, \dots, N \\
 u_{1,k} &\in U_1, k = 0, \dots, N-1 \\
 x_{1,N} &\in \Omega_1
 \end{aligned} \tag{10}$$

El controlador 2 resuelve el siguiente problema de optimización:

$$\begin{aligned}
 U_2^*(t) &= \operatorname{argmin}_{U_2} J_2(x_2(t), U_2, U_1^s(t)) \\
 x_{2,k+1} &= A_2 x_{2,k} + B_{22} u_{2,k} + B_{21} u_{1,k} \\
 x_{2,0} &= x_2(t) \\
 x_{2,k} &\in X_2, k = 0, \dots, N \\
 u_{2,k} &\in U_2, k = 0, \dots, N-1 \\
 x_{2,N} &\in \Omega_1
 \end{aligned} \tag{11}$$

En ambos problemas de optimización la variable de decisión es  $U_i$  mientras que la trayectoria de entrada vecina  $U_{ni}$  es fija.

3. Cada controlador  $i$  minimiza  $J_i$  optimizando la entrada vecina asumiendo que aplica la trayectoria de entrada calculada en el problema de optimización anterior  $U_i^*$ . El controlador 1 resuelve el siguiente problema de optimización.

$$\begin{aligned}
 U_2^w(t) &= \operatorname{argmin}_{U_2} J_1(x_1(t), U_1^*(t), U_2) \\
 x_{1,k+1} &= A_1 x_{1,k} + B_{11} u_{1,k} + B_{12} u_{2,k} \\
 x_{1,0} &= x_1(t) \\
 x_{1,k} &\in X_1, k = 0, \dots, N \\
 u_{1,k} &\in U_1, k = 0, \dots, N-1 \\
 x_{1,N} &\in \Omega_1
 \end{aligned} \tag{12}$$

El controlador 2 resuelve el siguiente problema:

$$\begin{aligned}
 U_1^w(t) &= \operatorname{argmin}_{U_1} J_2(x_2(t), U_2^*(t), U_1) \\
 x_{2,k+1} &= A_2 x_{2,k} + B_{22} u_{2,k} + B_{21} u_{1,k} \\
 x_{2,0} &= x_2(t) \\
 x_{2,k} &\in X_2, k = 0, \dots, N \\
 u_{2,k} &\in U_2, k = 0, \dots, N-1 \\
 x_{2,N} &\in \Omega_1
 \end{aligned} \tag{13}$$

En este problema de optimización, la variable de decisión es  $U_{ni}$  mientras que la trayectoria de entrada  $U_i$  es fija. Solucionando este problema de optimización, el controlador  $i$  define una trayectoria de entrada para su vecino que optimiza su función de costo local  $J_i$ .

4. Ambos controladores se comunican. El controlador 1 envía  $U_1^*(t)$  y  $U_2^w(t)$  al controlador 2 y recibe  $U_2^*(t)$  y  $U_1^w(t)$ .
5. Cada controlador evalúa la función de coste local  $J_1$  para cada una de las nueve diferentes combinaciones posibles; eso es  $U_1 \in (U_1^s(t), U_1^w(t), U_1^*(t))$  y  $U_2 \in (U_2^s(t), U_2^w(t), U_2^*(t))$ .
6. Ambos controladores se comunican y comparten la información del valor de la función de costo local para cada posible combinación de trayectorias de entrada. En esta etapa, ambos controladores reciben suficiente información para tomar una decisión cooperativa.
7. Cada controlador aplica la trayectoria de entrada que minimiza  $J = J_1 + J_2$ .
8. Debido a que ambos controladores tienen acceso a la misma información después del segundo ciclo de comunicación, ambos controladores eligen los mismos conjuntos de entrada óptimos. El conjunto elegido de trayectorias de entrada se denota:  $U_1^d(t), U_2^d(t)$ .
9. Se aplica la primera entrada de cada secuencia óptima y se repite el procedimiento el siguiente periodo de muestreo.

Desde el punto de vista de la teoría de los juegos, en cada periodo de muestreo ambos controladores están jugando un juego cooperativo.

### 3. Problema a resolver

#### 3.1. Descripción del sistema y modelo matemático

El sistema utilizado para validar la metodología propuesta en este trabajo es una planta formada por cuatro depósitos de agua conectados tal y como se indica en la figura 1.

Existen dos depósitos inferiores (tanques 1 y 2) y dos superiores (tanques 3 y 4) que desaguan en los tanques inferiores. Los tanques se llenan con dos bombas que proporcionan los caudales  $q_1$  y  $q_2$  desde un depósito de drenaje situado en

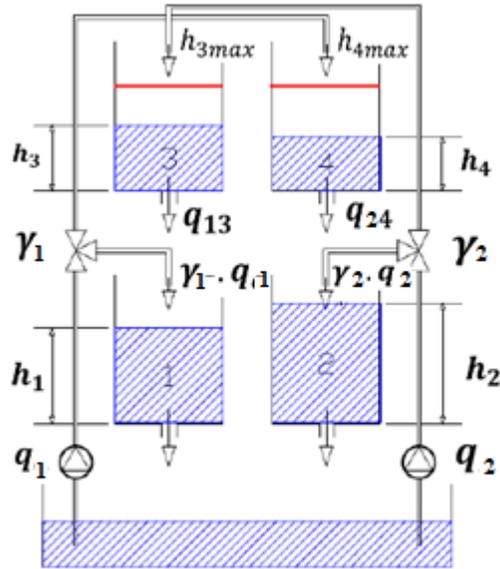


Figura 1: Planta formada por cuatro tanques

la parte más baja de la planta. Estos caudales se dividen en dos ramas, de forma que una fracción  $\gamma_1$  (o  $\gamma_2$ ) se envía por una de las ramas y el resto por la otra. Las fracciones  $\gamma_1$  y  $\gamma_2$  se fijan manualmente actuando sobre las válvulas ( $\gamma_1 = 0,3$  y  $\gamma_2 = 0,3$ ).

El modelo de la planta se puede obtener mediante balances de masas en cada depósito:

$$\frac{dh_1}{dt} = \frac{a_1}{S} \sqrt{2gh_1} + \frac{a_3}{S} \sqrt{2gh_3} + \frac{\gamma_1}{S} q_1 \quad (14)$$

$$\frac{dh_2}{dt} = \frac{a_2}{S} \sqrt{2gh_2} + \frac{a_4}{S} \sqrt{2gh_4} + \frac{\gamma_2}{S} q_2 \quad (15)$$

$$\frac{dh_3}{dt} = \frac{a_3}{S} \sqrt{2gh_3} + \frac{(1 - \gamma_2)}{S} q_1 \quad (16)$$

$$\frac{dh_4}{dt} = \frac{a_4}{S} \sqrt{2gh_4} + \frac{(1 - \gamma_1)}{S} q_2 \quad (17)$$

Donde la superficie de cada depósito es  $S = 0,03m^2$ ,  $g$  es la aceleración de la gravedad. La sección equivalente de los orificios de descarga ( $a_i$ ) es:

$$a_1 = 1,3104 - e4m^2; a_2 = 1,5074 - e4m^2; a_3 = 9,2673 - e5m^2; a_4 = 8,8164 - e5m^2$$

El modelo lineal en tiempo continuo que se utiliza como modelo de predicción es el siguiente:

$$\frac{dx}{dt} = A_c x + B_c u \quad (18)$$

$$y = C_c x \quad (19)$$

Donde:

$$A_c = \begin{bmatrix} \frac{1}{\tau_1} & 0 & \frac{1}{\tau_3} & 0 \\ 0 & \frac{1}{\tau_2} & 0 & \frac{1}{\tau_4} \\ 0 & 0 & \frac{-1}{\tau_3} & 0 \\ 0 & 0 & 0 & \frac{-1}{\tau_4} \end{bmatrix} \quad (20)$$

$$B_c = \begin{bmatrix} \frac{\gamma_1}{S} & 0 \\ 0 & \frac{\gamma_2}{S} \\ 0 & \frac{(1-\gamma_2)}{S} \\ \frac{(1-\gamma_1)}{S} & 0 \end{bmatrix} \quad (21)$$

$$C_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \quad (22)$$

$$\tau_i = \frac{S}{a_i} \sqrt{\frac{2h_i}{g}} \quad (23)$$

### 3.2. Problema de control y criterios de coordinación

Las variables manipuladas para el sistema de 4 tanques son los caudales proporcionados por las bombas  $q_1$ ,  $q_2$  y las variables controladas en este caso son los niveles de los depósitos inferiores  $h_1$ ,  $h_2$ . El sistema de cuatro tanques interconectados es interesante para la evaluación de estrategias de control distribuido por la interacción entre las variables y por la posibilidad de distinguir claramente subsistemas dentro de la planta. En los trabajos anteriores [1], [7].

Se han elegido estrategias de control centradas en el seguimiento de referencia y con ciertas consideraciones sobre la economía del proceso asociadas al coste de bombeo. En la implementación que se presenta en este trabajo, se desea aprovechar al máximo las posibilidades de los sistemas difusos de integrar criterios de operación de forma cualitativa. Para ello, se han establecido algunas condiciones de operación para hacer más interesante el problema de control distribuido.

Se desea minimizar el coste de bombeo (EB), es decir, el coste económico de la energía utilizada para bombear los caudales bombas  $q_a$  y  $q_b$  a los depósitos

$$EB = 0,04 * q_1 + 0,04 * q_b( EUR/d) \quad (24)$$

Se impone un tiempo de residencia mínimo en los depósitos inferiores

$$T_R = \frac{A * h}{Q} \quad (25)$$

Para evaluar este conjunto se compara el tiempo de residencia del tanque 1 ( $T_{R1}$ ) con el del tanque 2 ( $T_{R2}$ ) y se escoge el menor que se nombrará como  $T_R$ . Se define una altura máxima en los tanques superiores, existiendo una penalización económica si las alturas de los depósitos superiores ( $h_3$  y  $h_4$ ) sobrepasan el límite deseado.

Estas son condiciones hipotéticas que se han establecido para dar mayor complejidad a este caso de estudio, pero son condiciones que podrían existir en sistemas reales, donde se requiera un tiempo de residencia para garantizar la calidad de un producto o el almacenamiento de exceso de producto sea penalizado económicamente.

#### 4. Desarrollo de la propuesta

Para definir el sistema de inferencia difuso para la negociación de controladores es necesario tener un conocimiento del comportamiento del problema de control. Para obtenerlo se simuló la respuesta del sistema utilizando la estrategia del juego cooperativo. Se definieron tres escenarios de estudio que consisten en tres cambios de referencia en  $h_1$  y  $h_2$  en intervalos de tiempo que permiten alcanzar la condición de equilibrio (tabla 1).

En cada intervalo se calcula la energía de bombeo, el tiempo de residencia en los depósitos  $T_R$  y las alturas de los depósitos superiores  $h_3$  y  $h_4$ , cuando se ha alcanzado la referencia deseada para  $h_1$  y  $h_2$ . Estos valores se utilizaron para determinar los límites sobre el tiempo de residencia y la altura de los depósitos superiores y para obtener información del sistema a partir del análisis de la respuesta. En la tabla 2 se presentan los valores de las condiciones de operación para el caso 2. Con esta información se definieron los conjuntos difusos que describen de forma cualitativa el comportamiento del sistema en función de los criterios seleccionados. Estos son los antecedentes de las reglas difusas.

	Referencias	Inicial 200s 1700s		
Caso 1	$h_1(m)$	0.10	0.13	0.15
	$h_2(m)$	0.15	0.17	0.19
Caso 2	$h_1(m)$	0.10	0.20	0.10
	$h_2(m)$	0.15	0.20	0.15

Tabla 1: Cambios de referencia en los escenarios seleccionados para estudiar el comportamiento del sistema

	Referencias $h_1 = 0,1; h_2 = 0,15$ $h_1 = 0,2; h_2 = 0,25$ $h_1 = 0,1; h_2 = 0,15$		
$h_3(m)$	0.043	0.125	0.042
$h_4(m)$	0.061	0.043	0.06
$q_1(m^3/s)$	8.16 e-5	13e-5	6.66e-5
$q_2(m^3/s)$	7.77e-5	7.77e-5	7.5e-5
$T_{R1} (s)$	19.7	23.6	21.2
$T_{R2} (s)$	24.2	36.0	24.7

Tabla 2: Resultados de la simulación para la obtención de reglas difusas

#### 4.1. Antecedentes y conjuntos difusos

En el caso de la energía de bombeo, se calcula la variación de la energía de bombeo respecto a la condición inicial para cada cambio de referencia:

$$EB_n = \frac{EB - EB_{ss}}{EB_{ss}} \quad (26)$$

De este análisis se obtienen las particiones que se muestran en la figura 2, para describir el efecto de los movimientos de las variables manipuladas  $q_1$  y  $q_2$  sobre la energía de bombeo.

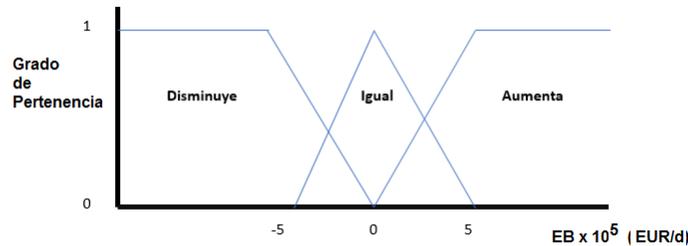


Figura 2: Conjuntos difusos definidos por la energía de bombeo ( $EB$ )

En el caso del tiempo de residencia  $T_R$ , se ha establecido un requerimiento de  $T_R > 20$  segundos a partir del análisis del comportamiento del sistema. Este sería el tiempo necesario de almacenamiento para que el producto cumpla ciertas especificaciones. Los conjuntos difusos estarían definidos simplemente como: cumple, no cumple. Se toma como variable de decisión la menor entre  $T_{R1}$  y  $T_{R2}$ . En la figura 3 se muestran las particiones.

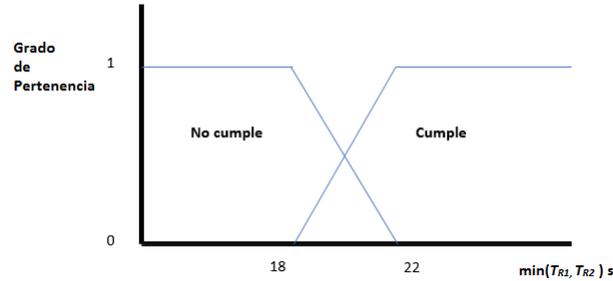


Figura 3: Conjuntos difusos definidos para el tiempo de residencia

En el caso de la altura de los depósitos superiores, se establece a través de un análisis de las simulaciones previas, un límite de 0.12m. Alturas superiores a ese valor son penalizadas, como producto en exceso almacenado. Se toma como variable de decisión la mayor entre  $h_3$  y  $h_4$ . En la figura 4 se presentan las particiones para esta variable de decisión.

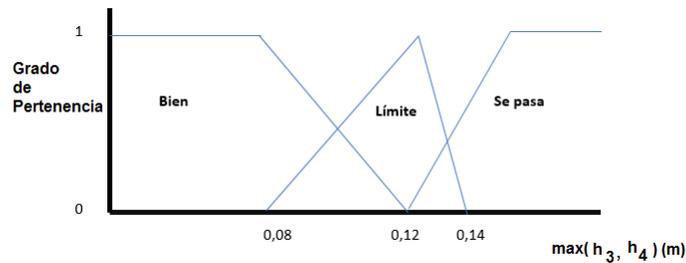


Figura 4: Conjuntos difusos definidos por las alturas  $h_3$  y  $h_4$

Los antecedentes se utilizan para generar las reglas difusas que se utilizan para decidir qué señal de control aplicar en cada instante de muestreo. Se consideran 10 reglas difusas que permiten decidir sobre la bondad de la solución evaluada. Una solución buena, es una solución que cumple con los requisitos de  $T_R$  y de altura de los depósitos superiores, y cuya energía de bombeo (EB) disminuye o es igual a la del punto de operación. En caso contrario se considera una

solución deficiente. Entonces, las reglas difusas se usan para asignar un valor a la calidad de cada posible señal de control, clasificándolas como: buena, aceptable, mala, asignando un valor a su importancia en el cálculo de la señal de control a enviar a la planta de 1 para las soluciones buenas, 0.5 para las una aceptables y 0.3 para las malas.

Estas reglas son:

1. Si  $T_R$  cumple,  $EB$  disminuye y  $max(h_3, h_4)$  bien la solución es buena
2. Si  $T_R$  cumple,  $EB$  es igual y  $max(h_3, h_4)$  bien la solución es buena.
3. Si  $T_R$  cumple,  $EB$  aumenta y  $max(h_3, h_4)$  bien la solución es aceptable
4. Si  $T_R$  cumple,  $EB$  disminuye y  $max(h_3, h_4)$  límite la solución es aceptable
5. Si  $T_R$  cumple,  $EB$  es igual y  $max(h_3, h_4)$  límite la solución es aceptable
6. Si  $T_R$  cumple,  $EB$  aumenta y  $max(h_3, h_4)$  límite es mala
7. Si  $T_R$  cumple,  $EB$  disminuye y  $max(h_3, h_4)$  se pasa es aceptable
8. Si  $T_R$  cumple,  $EB$  es igual y  $max(h_3, h_4)$  se pasa es aceptable
9. Si  $T_R$  cumple,  $EB$  aumenta y  $max(h_3, h_4)$  se pasa es mala
10. Si  $T_R$  no cumple esta solución es mala

La importancia de la solución evaluada en cada regla se calcula como el producto de los grados de pertenencia multiplicado por un factor de ponderación como se ha descrito anteriormente, donde  $j$  se refiere a cada señal de control posible e  $i$  se refiere a la reglas difusas.

$$ImportanciaR_i(j) = ponderacion * fp(TR) \cdot fp(EB) \cdot fp(H) \quad (27)$$

La importancia total de esa solución será la suma de las importancias de cada regla para esa solución, la cual se normaliza después teniendo en cuenta la importancia de las demás soluciones. Finalmente, la señal de control a enviar es la siguiente:

$$U_{aplicar} = \sum_{j=1}^j U(j) \cdot ImportanciaTotal(j) \quad (28)$$

## 5. Resultados

En este apartado se compara el resultado obtenido al controlar el sistema de cuatro tanques interconectados utilizando el sistema de inferencia difuso para realizar la negociación entre los controladores que determina la señal de control óptima. Estos resultados se comparan con lo obtenido en el método de juego cooperativo descrito anteriormente. Se realizaron simulaciones considerando cambios de referencia y perturbaciones en distintas zonas de la planta. Estos resultados se presentan en las figuras 5 a 7. Los cambios de referencia corresponden al caso 2, las perturbaciones son una obstrucción de los orificios de salida de

los tanques (reducción del 10 %) que ocurre a los 950 s, y la segunda representa una pérdida de presión a la salida de las bombas que reduce el caudal en 10 % (ocurre a los 2450 s).

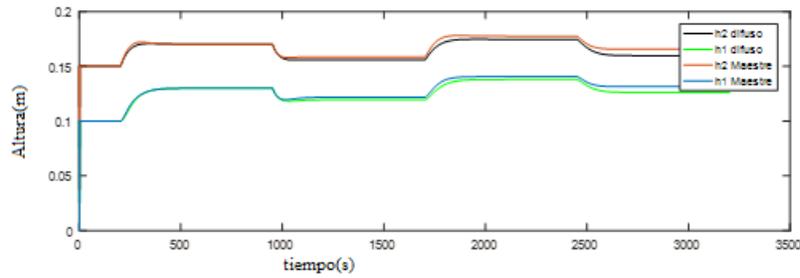


Figura 5: Evolución de las alturas  $h_1$  y  $h_2$

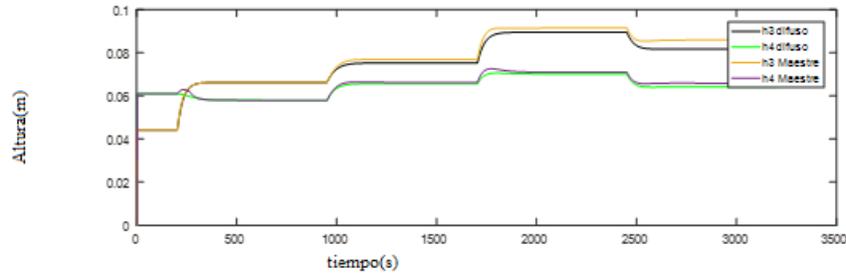


Figura 6: Evolución de las alturas  $h_3$  y  $h_4$

En las figuras 5 y 6 se puede observar que el comportamiento con ambas técnicas presenta ciertas diferencias generadas por la introducción de los criterios adicionales de costes y restricciones. Las señales de control obtenidas con el método de negociación difusa son de menor magnitud, por la minimización de los costes de bombeo y la penalización sobre  $h_3$  y  $h_4$ . La respuesta a perturbaciones es aceptable, aunque algunas producen un error en edo. estacionario debido a la imposibilidad de ambos algoritmos para rechazarlas completamente.

## 6. Conclusiones

Se ha implementado un algoritmo alternativo al de juego cooperativo que utiliza la lógica borrosa para introducir criterios de operación en la negociación,

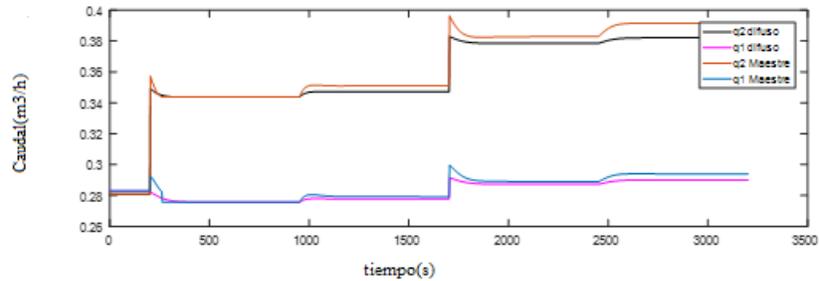


Figura 7: Señales de control obtenidas

restricciones y criterios económicos aplicado al caso concreto de un sistema de cuatro tanques acoplados. Esta técnica presenta la ventaja frente al algoritmo del juego cooperativo de no necesitar evaluar los costes para las distintas combinaciones de soluciones obtenidas a través del algoritmo de Maestre [7], lo cual podría ser un trabajo costoso si se tuviera un número elevado de controladores. Además, el sistema propuesto permite aplicar señales de control que de algún modo son una “mezcla” de la solución que propone cada uno de los controladores locales, minimizando los costes sin deteriorar el desempeño. Para comprobar el desempeño de este método se realizaron diferentes simulaciones en distintos escenarios, añadiendo en algunos de ellos perturbaciones lo cual supone una novedad frente a trabajos anteriores. En las simulaciones se pudo ver como este método mejoraba los resultados obtenidos anteriormente en cuanto a coste y satisfacción de las restricciones y criterios económicos con otras técnicas sobretodo cuando se introducían perturbaciones. Los resultados son satisfactorios obteniéndose un buen seguimiento de la referencia y rechazo adecuado a perturbaciones con menores esfuerzos de control. La técnica es prometedora para extenderla a plantas más complejas debido a su simplicidad.

**Agradecimientos.** Los autores agradecen la financiación MINECO a través del proyecto DPI2015-67341-C2-1 y de la Fundación Samuel Solórzano (FS/21 - 2015).

## Referencias

1. I. Alvarado, D. Limon, D. Muñoz De La Peña, J. M. Maestre, M. A. Ridaó, H. Scheu, W. Marquardt, R. R. Negenborn, B. De Schutter, F. Valencia, J. Espinosa, and I. Alvarado, “A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark \* A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark,” *Journal of Process Control*, vol. 21, no. 5, pp. 800–815, 2011. [Online]. Available: [http://www.dsc.tudelft.nlhttp://pub.deschutter.info/abs/11\\_025.html](http://www.dsc.tudelft.nlhttp://pub.deschutter.info/abs/11_025.html)

2. E. Fernandez-Camacho and C. Bordons-Alba, *Model Predictive Control in the Process Industry*. Springer London, 1995. [Online]. Available: <https://books.google.es/books?hl=es&lr=&id=hUDhBwAAQBAJ&oi=fnd&pg=PA1&dq=Model+Predictive+Control+in+the+Process+Industry&ots=B17xchhAjq&sig=BW1nEGw6UK8Bt8ddJtmhd4kjNbo#v=onepage&q=Model%20Predictive%20Control%20in%20the%20Process%20Industry&f=false>
3. E. P. Gatzke, E. S. Meadows, C. Wang, and F. J. Doyle, "Model based control of a four-tank system," *Computers & Chemical Engineering*, vol. 24, no. 2-7, pp. 1503–1509, 7 2000. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S009813540000555X>
4. M. J. Grimble and U. A. Michael Johnson, "Advanced Textbooks in Control and Signal Processing Series editors." [Online]. Available: <https://link.springer.com/content/pdf/10.1007/978-3-319-24853-0.pdf>
5. K. H. Johansson, "The Quadruple-Tank Process: A Multivariable Laboratory Process with an Adjustable Zero," *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY*, vol. 8, no. 3, 2000. [Online]. Available: <http://www.diva-portal.org/smash/get/diva2:495784/FULLTEXT01.pdf>
6. D. Limon, I. Alvarado, T. Alamo, and E. Camacho, "MPC for tracking piecewise constant references for constrained linear systems," *Automatica*, vol. 44, no. 9, pp. 2382–2387, 9 2008. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0005109808001106>
7. J. M. Maestre, D. M. d. l. Peña, and E. F. Camacho, "Distributed model predictive control based on a cooperative game," *Optimal Control Applications and Methods*, vol. 32, no. 2, pp. 153–176, 3 2011. [Online]. Available: <https://onlinelibrary.wiley.com/doi/full/10.1002/oca.940>
8. M. Mercangöz, "Distributed model predictive control of an experimental four-tank system," *Journal of Process Control*, vol. 17, no. 3, pp. 297–308, 3 2007. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0959152406001387>
9. J. B. Rawlings and D. Q. Mayne, "Postface to " Model Predictive Control: Theory and Design " Chapter 2. Model Predictive Control — Regulation," 2012. [Online]. Available: <https://pdfs.semanticscholar.org/74ec/aa837079d2a4769eee4ac7951e33d52c8d2a.pdf>
10. H. Scheu and W. Marquardt, "Sensitivity-based coordination in distributed model predictive control," *Journal of Process Control*, vol. 21, no. 5, pp. 715–728, 6 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0959152411000163>

# Sistema Conexionista Aplicado a la Identificación Biométrica

Michael Galdámez<sup>1</sup> y Angélica González Arrieta<sup>1</sup>

Departamento de Informática y Automática, Facultad de Ciencias  
Plaza de los Caídos s/n, 37008, Salamanca, España  
{michael.galdamez, angelica}@usal.es

**Resumen** La detección y el reconocimiento de iris es un tema que se está tratando ampliamente debido a la gran cantidad de aplicaciones que se le puede dar y a su confiabilidad. El reconocimiento de iris tiene la ventaja que, a diferencia de otros sistemas biométricos, es un órgano interno que está protegido contra los daños y el desgaste de los años. Además, con la tecnología disponible, es fácil tomar las imágenes con suficiente resolución y sin que la persona se percate de ello. El presente trabajo tiene como objetivo construir un sistema capaz de detectar el ojo de una persona presente en una imagen, extraer las características del iris y luego determinar la identidad de dicho individuo. Para lo cual se utiliza el algoritmo Haar-Cascade para la detección, se implementan algunas técnicas de procesamiento de imágenes, como la conversión a escala de grises, un filtro gaussiano, la binarización de la imagen, entre otras. Se extraen las características por medio de la aplicación del algoritmo de análisis de componentes principales (ACP), construyendo un arreglo de eigenvalues los cuales sirven como entrada de una red neuronal que se encargará de reconocer al individuo.

**Keywords:** Iris recognition, principal components analysis, neural networks, eigenvalue, haar-cascade, hausdorff distance

## 1. Introducción

A lo largo de la historia el ser humano ha mantenido una obsesión por la seguridad, se han desarrollado diversas técnicas para la identificación y así evitar el acceso de personas no deseadas a información personal o a lugares con acceso restringido, lo que en una sociedad como la actual en la que el acceso a la información es tan fácil, la seguridad viene a ser un aspecto fundamental. Por lo tanto es necesario desarrollar sistemas que nos ayuden a proteger al usuario.

Es por eso que se comienza a investigar nuevos sistemas que puedan llegar a garantizar la perdurabilidad de la seguridad, lo que da lugar al uso de los rasgos de las personas en conjunto con fundamentos matemáticos para el reconocimiento basado en características físicas a lo cual se le llama *biometría*.

La tecnología de reconocimiento de iris es sin duda una de las más confiables, debido a las características únicas, estables y accesibles del patrón del iris. Se han desarrollado varios sistemas de reconocimiento de iris (comerciales y no comerciales). Por lo antes mencionado, se propone el desarrollo de un sistema de identificación de personas mediante el reconocimiento de iris.

Este trabajo se ha centrado en la implementación de algunos algoritmos para el reconocimiento del iris, y esta organizado de la siguiente manera: En la Sec. 2 se hace un repaso sobre algunos conceptos básicos del reconocimiento de iris. La Sec. 3 describe las estrategias propuestas en las investigaciones para el problema del reconocimiento de iris y la estructura de un sistema típico. En la Sec. 4 se encuentra el grueso de la propuesta a nivel teórico, los detalles del método propuesto y el sistema real desarrollado. En la Sec. 5 se presentan los resultados extraídos de la investigación de este trabajo en la tarea de reconocimiento de iris y, finalmente en la Sec. 6 se presentan las conclusiones y las líneas de trabajo futuro.

## 2. Método Propuesto

En esta sección se describe tanto los algoritmos propuestos como las técnicas de pre-procesamiento y diferentes métodos de extracción de características que serían utilizados.

### 2.1. Detección

Para realizar la detección de ojos se utilizó la librería OpenCV con la cuál se implementó el algoritmo *haar-cascade* el cual se basa en el entrenamiento de un clasificador usando un número de imágenes positivas que representan el objeto a reconocer y un número de imágenes negativas que representan objetos o características que no se detectarán (Las imágenes negativas podrían ser cualquier cosa que no represente el objeto a reconocer) [1]. La implementación de OpenCV para el entrenamiento del detector, se denomina *HaarTraining*. Se deben de especificar algunos parámetros antes de iniciar el entrenamiento.

En primer lugar, se debe de especificar la cantidad de muestras positivas y negativas que se utilizan para realizar el entrenamiento de las capas de la cascada. Antes de comenzar el entrenamiento de cada clasificador fuerte, el algoritmo se asegura de contar con la cantidad de imágenes necesaria. Las imágenes deben de haber pasado las etapas anteriores de la cascada, por lo que a medida que se avanza en la cascada, se va depurando de imágenes negativas. Para el entrenamiento seleccionamos 5000 imágenes de la base de datos de ojos UBIRIS, las cuales fueron utilizadas como muestras positivas. Así mismo seleccionamos 3000 imágenes negativas. En segundo lugar, se debe de especificar el número de etapas a entrenar, en las pruebas que se llevaron a cabo se utilizó una cantidad de 30 etapas. Esta cantidad debe de decidirse en conjunto con la tasa de detección

( $TD_i$ ) y la tasa de falsos positivos ( $TFP_i$ ) de cada etapa  $i$ , ya que la  $TD$  y  $TFP$  general de la cascada dependen de estos tres parámetros.

En nuestro entrenamiento fijamos  $TD_i$  en 99,9% y  $TFP_i$  en 60%. Teniendo en cuenta que la cascada constará de 30 etapas, la tasa de detección general será aproximadamente de  $0,999^{30} \approx 0,9704 = 97,04\%$  y la tasa de falsos positivos será aproximadamente  $0,60^{30} \approx 0,000000221 = 0,0000221\%$ . Una vez la aplicación haya finalizado su trabajo la cascada entrenada se guardará en un archivo *cascade.xml*.

## 2.2. Tratamiento de la Imagen

El primer paso para conseguir nuestros objetivos es obtener las características que nos permitan diferencias entre las distintas clases de imágenes almacenadas. Las imágenes tratadas en el sistema serán obtenidas de la base de datos UBIRIS, ya que esta base de datos presenta diferentes sesiones de fotos del ojo de un individuo, que luego se intentará identificar, para la correcta obtención de las características de las imágenes se realizarán diferentes operaciones.

**Escala de Grises.** En primer lugar la imagen debe de convertirse en un formato más apropiado para extraer las características. Se utiliza RGB, de esta manera se asigna una intensidad a cada uno de los colores RGB (Rojo, verde y azul), así cada píxel en una imagen se representa mediante un valor que identifique la intensidad de cada uno de estos tres colores, con un único valor entre 0 y 255, el cual representa que tan oscuro es el mismo. Para convertir un píxel de una imagen en color a uno en escala de grises se suele hacer un promedio ponderado de la intensidad de cada uno de los tres colores en donde a cada color se le asigna un peso. La suma de los tres pesos debe ser igual a 1 [2]. Los pesos utilizados por OpenCV son:  $gris = 0,299 \times rojo + 0,587 \times verde + 0,114 \times azul$ . El resultado de aplicar la escala de grises es el que se observa en la Fig. 1.



(a) Imagen original



(b) Imagen en escala de grises

Figura 1: Conversión a escala de grises

**Filtrado de Imagen.** Para poder quitar ruido, es necesario realizar algún tipo de filtrado a la imagen, una forma de hacer esto es reemplazar cada píxel por la media del valor de los píxeles de alrededor, de esta forma las variaciones rápidas de intensidad pueden ser suavizadas y reemplazadas por una transición más gradual, en otras palabras el filtrado Gaussiano es igual que un promediado pero ponderado. Esta ponderación se hace siguiendo la campana de Gauss como se observa en la Fig. 2, con esto se consigue dar más importancia a los píxeles que están más cerca del centro de los que están más alejados.

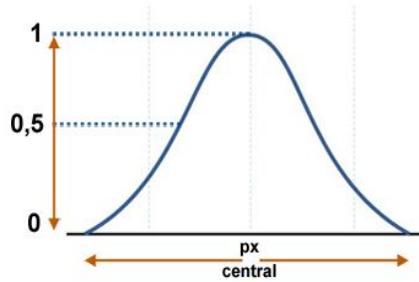


Figura 2: Función Gaussiana

La función de OpenCV elegida para realizar este suavizado ha sido *GaussianBlur* [2], la cual tiene en cuenta el peso de los píxeles más cercanos que los alejados.

$$G(x, y) = \frac{1}{2 \times \pi \times \sigma^2} \times e^{\frac{-x^2+y^2}{2 \times \sigma^2}} \quad (1)$$

Donde la  $x$  es la distancia desde el origen en el eje horizontal, e  $y$  es la distancia desde el origen en el eje vertical y  $\sigma$  es la desviación estándar de la distribución Gaussiana. La Fig. 3 muestra el resultado de aplicar la función *GaussianBlur* de OpenCV sobre una imagen en escala de grises.



(a) Imagen en escala de grises



(b) Imagen suavizada

Figura 3: Imagen con un filtro *GaussianBlur* aplicado

**Binarización.** Al proceso de conversión de una imagen en escala de grises a una imagen en blanco y negro se le conoce como binarización, este es realmente útil en las tareas de reconocimiento ya que separa las regiones correspondientes del objeto que se desea analizar, esta separación se basa en la variación de intensidad entre los píxeles del objeto y los píxeles del fondo, para diferenciar los píxeles que nos interesan del resto (que eventualmente será rechazado), realizamos una comparación de la intensidad de cada píxel con respecto a un valor umbral.

Una vez que hemos separado correctamente los píxeles importantes, podemos establecerlos con un valor determinado para identificarlos, es decir asignarles un valor 0 (negro), 255 (blanco). En OpenCV existe la función *Threshold* para realizar este tipo de operación [2].

$$dst(x, y) = \begin{cases} 255 & \text{si } I(x, y) < T(x, y) \\ 0 & \text{en cualquier otro caso} \end{cases} \quad (2)$$

Esta ecuación se implementó, sin embargo cuando las imágenes presentan poca luz, se obtiene un mejor contorno de la misma con una variación de esta.

$$dst(x, y) = \begin{cases} 0 & \text{si } I(x, y) < T(x, y) \\ 255 & \text{en cualquier otro caso} \end{cases} \quad (3)$$

Normalmente, la diferencia entre estas dos funciones se encuentra en cómo elige cada una el valor umbral  $T(x, y)$ .

El resultado obtenido sobre la imagen suavizada se puede observar en la Fig. 4.



(a) Imagen suavizada



(b) Imagen binarizada

Figura 4: Binarización de la imagen

**Detección de Bordes Canny.** Luego de la binarización de la imagen, es necesario detectar los bordes, este filtro es opuesto al GaussianBlur, en lugar de suavizar la imagen para quitar ruido, se buscan los bordes y formas de la imagen. El algoritmo Canny pasa por dos etapas: reducción de ruido y encontrar la intensidad del gradiente de la imagen. El resultado de la aplicación de este algoritmo se muestra en la Fig. 5.



Figura 5: Detección de bordes Canny de la imagen

**Ecualización del Histograma.** El histograma puede considerarse como la representación gráfica del valor de píxeles frente al recuento de píxeles. La ecualización del histograma es el ajuste del contraste usando el histograma de la imagen, este se realiza en las imágenes demasiado oscuras o demasiado brillantes con el fin de mejorar la calidad de imagen y para mejorar el rendimiento del reconocimiento. Se modifica el rango dinámico (rango de contraste) de la imagen y, como resultado, algunas características se hacen más evidentes [2]. La Fig. 6 muestra lo que realmente hacemos con la ecualización del histograma.

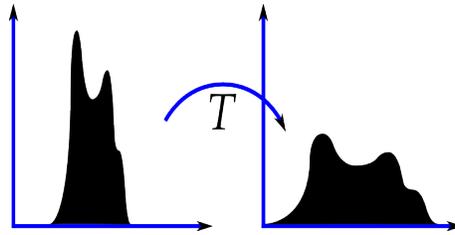


Figura 6: Efecto de la ecualización del histograma

La ecualización del histograma mejora el contraste de una imagen al estirar su rango de intensidad. En la Fig. 7 se observa, la primera trama representa el histograma de la imagen de bajo contraste y en la segunda, el histograma después de la ecualización. Por lo tanto, es más claro que la ecualización del histograma sólo estira el rango de intensidad. En la Fig. 7 se puede observar el resultado de aplicar la ecualización del histograma:

### 2.3. Extracción de Características

Finalmente, después de procesar adecuadamente la imagen, es necesario implementar el análisis de componentes principales (ACP), el cual es un procedimiento que extrae las características más importantes de un conjunto de datos. Una imagen, se puede representar utilizando un vector si el ancho y la altura de



(a) Imagen en escala de grises      (b) Imagen ecualizada

Figura 7: Imagen con ecualización del histograma

la imagen es de  $w \times h$  píxeles respectivamente. Cada valor numérico del píxel, corresponderá a un elemento del vector. La construcción de dicho vector a partir de la imagen, se realiza por medio de una simple concatenación.

El objetivo de utilizar análisis de componente principales (ACP), consiste en reducir la dimensión de un conjunto de modo que el nuevo describa mejor los patrones típicos del conjunto total. Es decir la idea central del ACP consiste en encontrar un espacio de características, el cual capturará la mayor parte de la variación que se presenta en el conjunto de datos originales.

**Eigenvectors y Eigenvalues.** Un eigenvector como su nombre lo dice es un vector que si es multiplicado por una matriz, nos da como resultado un múltiplo entero del vector. Este valor entero es el correspondiente eigenvalue. Esta relación puede ser descrita por la ecuación  $M \times u = \lambda \times u$ , donde  $u$  es un eigenvector de la matriz  $M$  y  $\lambda$  el correspondiente eigenvalue [6].

**Descripción del Método.** El primer paso consiste en crear una matriz con las imágenes, luego es necesario obtener la media correspondiente a cada una de las variables del arreglo de datos, y llevar a cabo un ajuste de los datos, restando a cada uno la media correspondiente a su variable, lo siguiente es formar una matriz de covarianza, en seguida se procede a calcular los eigenvectors y los eigenvalues de la matriz de covarianza, una vez que se tienen estos, debemos construir un vector característico que es sólo otra manera de representar la matriz de eigenvectors organizando estos de mayor a menor a manera de columnas según eigenvalues de la matriz de covarianza.

Otra opción para llevar a cabo lo anterior consiste en despreciar los eigenvectors menos significativos (el realizar lo anterior se verá reflejado en una reducción de dimensiones, lo que implicará pérdida de datos, en este caso los menos significa-

tivos). Los pasos anteriores se llevan a cabo de forma transparente utilizando la librería OpenCV.

Se utiliza el algoritmo ACP para extraer las características de la imagen y esas características son definidas como un arreglo de eigenvalues que posteriormente se introducen en una red neuronal que utiliza una neurona de entrada por cada eigenvalue. Los eigenvalues son un valor por cada imagen en el conjunto de entrenamiento, esto sería como la distancia euclídea entre una imagen y la imagen de entrada a reconocer, se decidió utilizar los eigenvalues del ACP ya que en lugar de enviar píxeles y sus valores como entrada de la red neuronal es más eficiente y tiene un coste computacional menor reducir la imagen a un solo valor double.

En ACP primero se calculan los eigenvectors que viene a ser la distancia (similitud) entre cada pixel de la imagen de entrada contra las imágenes en la base de datos eso es una matriz de números, y el eigenvalue es como la distancia promedio de todos los píxeles en las imágenes de la base de datos contra la imagen de entrada. Es decir, de una matriz pasamos a tener un arreglo unidimensional, es por esto que se utilizan los eigenvalues y no los eigenvectors. Por tanto si para un sujeto se tienen 10 imágenes, el algoritmo ACP calculará un eigenvalue por cada una de esas 10 imágenes y se convertirá entonces en 10 neuronas de entrada para ese sujeto.

#### 2.4. Obtención de la Red Neuronal

Las redes neuronales normalmente están compuestas por una serie de capas de neuronas.

Como se puede observar en la Fig. 8 se tiene una serie de entradas de  $N$  variables, denominado  $x_1, x_2, \dots, x_n$ . A cada una de las líneas que conectan las entradas con la neurona se le define un peso, que se denota como  $W_k1, W_k2, \dots, W_kn$ , respectivamente. El umbral de la neurona está representado por  $\theta_k$  y la activación está dada por la función  $F(x)$ . Se pueden distinguir tres tipos de capas:

- Entrada: estas capas reciben la información desde el exterior.
- Salida: estas son las encargadas de enviar la información hacia el exterior.
- Ocultas: estas capas sirven para procesar información y comunicarse con otras capas.

La salida de cada neurona es como la suma promedio de cada una de sus entradas antes de la multiplicación por su peso correspondiente. Ahora bien, si dicho peso es positivo, entonces podemos decir que la entrada es excitadora; en caso contrario será inhibitoria. La salida que se genera es filtrada por una función de activación  $F(x)$ , que bien podría tratarse de una función salto  $[0,1]$  desplazada del origen por cierto umbral  $\theta_k$ . La función empleada en este caso es la sigmoidea, la razón es de conveniencia numérica, ya que es una función  $n$ -diferenciable [3].

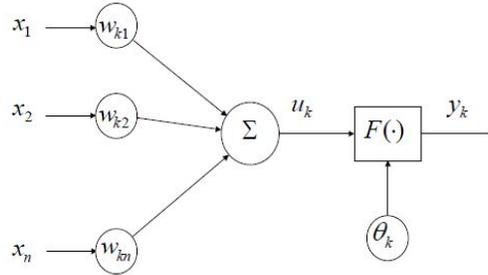


Figura 8: Red neuronal

El trabajo que llevará a cabo la red neuronal, está directamente relacionado con el número de neuronas de salida, en el caso de la tarea de reconocimiento del iris se establece una neurona por cada sujeto que se encuentra en la base de datos, y se elige como función de activación, la función sigmoidea, la cuál está definida por:

$$f(x) = \frac{1}{(1 + e^{-x})} \quad (4)$$

La capa oculta se encuentra entre las capas de entrada y salida. En una red, cuando solo existe una sola capa de entrada y una sola capa de salida, se le llama red de una sola capa. Si hay una o más capas ocultas, se les llama redes multicapa. Existen diferentes tipos de arquitectura de redes neuronales, para este trabajo se implemento una red *feedforward*. Las redes de propagación hacia delante o feedforward, las neuronas se organizan en capas. Las neuronas de una capa tienen que obtener la respuesta de la capa anterior y alimentar así su salida que es la conexión a la siguiente capa. En este tipo de redes no se permiten las conexiones entre las mismas capas o capas anteriores.

**Funcionamiento de la Red Neuronal.** Como hemos mencionado antes, las redes neuronales se utilizan para tareas complejas en una gran variedad de campos de aplicación, entre las que encontramos el reconocimiento del iris del ojo, en esta solución se encuentra una red neuronal que identifica a cada sujeto en la base de datos.

Una vez se extraen las características de la imagen, esas son definidas como un arreglo de eigenvalues, que posteriormente se introducen en una red neuronal que utiliza una neurona de entrada por cada eigenvalue para entrenar la red neuronal, en el algoritmo de entrenamiento los vectores de valores que pertenecen a una persona, se utilizan como ejemplos positivos para la neurona de dicho sujeto devolviendo 1 como salida de la neurona asignada a ese sujeto y 0 en el resto de neuronas. En la Fig. 9 se puede observar un diagrama del entrenamiento de la red neuronal, y en la Fig. 10 se muestra una simulación de entrenamiento de la misma.

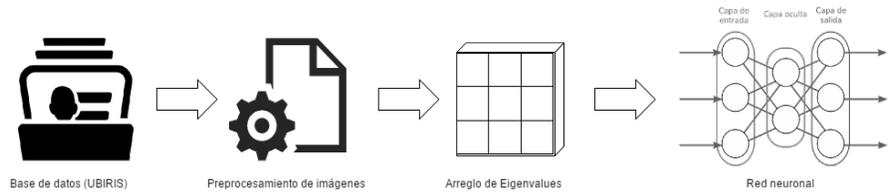


Figura9: Entrenamiento de la red neuronal

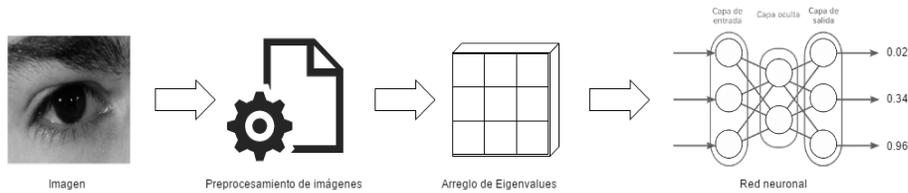


Figura10: Simulación de la red neuronal

Cuando se carga una imagen para el reconocimiento, se calculan los eigenvalues de la imagen, estos se introducen en la red neuronal, las salidas de las diferentes neuronas se cotejan, y si la salida con el valor máximo supera el nivel de umbral definido, entonces se decide que el iris del ojo corresponde al sujeto asignado a la neurona con el índice activado.

## 2.5. Funcionamiento del Sistema de Reconocimiento

En las secciones anteriores hemos expuesto los distintos pasos que se llevan a cabo, como ser el tratamiento de la imagen, el procedimiento para obtener el arreglo de eigenvalues de una colección de imágenes, el entrenamiento de la red de neuronal. A continuación se presenta de manera íntegra el procedimiento aplicado en el sistema.

**Obtención de Imágenes.** El primer paso es la obtención de imágenes de sujetos, para que luego de obtener un conjunto de sujetos, reconocerlos. Como hemos mencionado anteriormente se utilizó la base de datos UBIRIS la cuál contiene varias sesiones de fotos de los sujetos, dividimos la base de datos en 80 % que se utilizó para el entrenamiento y 20 % que se utilizaron para imágenes de prueba.

**Procesamiento de Imágenes.** Una vez que se poseen las imágenes, ya es posible realizar las identificaciones para lo cual se comienza con la carga de

la imagen del posible sujeto a reconocer. Con la imagen cargada se procede a convertir en escala de grises, se aplican los filtros, la binarización, detección de borde, se extrae el ojo y se ajusta su tamaño, se remueve el fondo aplicando una máscara en forma de elipse negro alrededor del ojo identificado, para finalmente ecualizar la imagen mejorando así la iluminación.

El proceso mostrado en la Fig. 11 se realiza para todas las imágenes de la base de datos, también para la imagen cargada al momento de realizar el reconocimiento del iris.



Figura 11: Diagrama del procesamiento de una imagen

**Reconocimiento del Sujeto.** Una vez se obtiene el conjunto de imágenes procesadas, se construye un arreglo con sus respectivas etiquetas, el cual indicará a quien corresponde cada una de las imágenes, a la cual se calculan los eigenectores y los eigenvalues aplicando el algoritmo de análisis de componentes principales (ACP), pero en lugar de enviar píxeles y sus valores como entrada de la red neuronal se tomó la decisión de reducir la imagen a un solo valor double, se decidió establecer un valor umbral de 1000 el cual muestra una similitud entre dos imágenes del 90 %. Es decir que se obtienen los eigenvalues de la colección de imágenes almacenadas en la base de datos, y los eigenvalues de la imagen cargada, se comparan ambos con la medida de similitud, lo cual tiene como resultado un vector de similitudes de la imagen cargada con respecto al conjunto de la base de datos, se evalúa el vector, se obtiene el valor y se compara con el umbral definido en caso de superarlo devuelve la etiqueta del sujeto reconocido, de lo contrario no se ha reconocido.

**Resumen de los Pasos Aplicados.** El reconocimiento del iris se puede resumir en los siguientes pasos:

1. Obtener una base de datos de ojos que contenga varias imágenes de los sujetos a reconocer.
2. Seleccionar las imágenes de cada persona con alguna variación en la distancia, iluminación, etc. Las cuales sirven como conjunto de entrenamiento ( $M$ ).

3. Calcular los eigenvectores y eigenvalues, los eigenvalues son un valor por cada imagen en el conjunto de entrenamiento.
4. Crear una red neuronal, con una neurona de salida por cada sujeto.
5. Utilizando las imágenes de cada sujeto como ejemplos positivos de sus neuronas y ejemplos negativos para el resto, se entrena la red neuronal.
6. Para la imagen cargada obtener los eigenvectores y sus respectivos eigenvalues.
7. Utilizar el eigenvalue como entrada a la red neuronal.
8. Seleccionar la neurona con el valor más alto, si la salida de la neurona seleccionada es mayor al umbral definido, se establece como reconocido sino se presenta como no reconocido.

Como hemos expuesto hasta el momento los algoritmos y los métodos utilizados presentan un porcentaje de precisión bastante elevado, estos nos ayudan a tener un sistema altamente aceptable, bajo condiciones estándares de iluminación. Con la intención de brindar más potencia al sistema, se implementa la técnica de distancia Hausdorff la cual nos permite realizar la comparación de dos figuras, partiendo de las distancias entre los píxeles que las forman. Este método no utiliza los eigenvalues, sino que trabaja sobre los mapas de bordes de la imagen cargada y el conjunto de entrenamiento.

## 2.6. Implementación de la Distancia Hausdorff

La medida de distancia Hausdorff implementa la forma de comparación en los mapas de borde, la ventaja de utilizar bordes para comparar dos figuras es que esta representación es sumamente eficiente y presenta resultados precisos frente al cambio de iluminación. Basados en esta medida, el efecto que se obtendrá gracias a la implementación de un algoritmo de detección de bordes será significativo [5]. En la Fig. 12 se muestra como funciona la implementación del algoritmo.

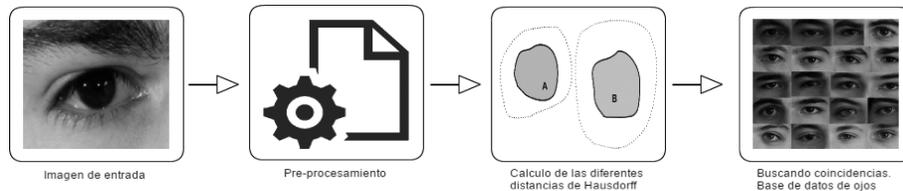


Figura 12: Diagrama implementación distancia de Hausdorff

La implementación del algoritmo consiste en retirar el fondo de la imagen, como se realizó en el tratamiento de la imagen original, luego de aplicar la máscara y la ecualización se procede a obtener los bordes, una vez aplicado este filtro se invierte la imagen para poder trabajar con un fondo blanco, finalmente se binariza la imagen, un proceso parecido al expuesto en la sección anterior.

Una vez se obtienen las figuras, se realiza una comparación, obteniendo la distancia de Hausdorff lo que quiere decir comparando píxeles contra píxeles y de esta manera podemos saber que tan similares son las dos figuras dando como resultado una colección de valores en la cual se encuentra la distancia de la imagen cargada en relación con cada imagen en la base de datos, la figura que tiene la menor distancia relativa es presentada como una candidata posible, si este valor logra superar el umbral definido entonces se reconoce el sujeto. El algoritmo Hausdorff es presentado como una alternativa a la red neuronal, si el procedimiento identifica que el sujeto es el mismo, incluso sin lograr superar los umbrales definidos, se acepta que la imagen que se carga pertenece al sujeto que la red neuronal identificó.

### 3. Resultados

Sobre la propuesta presentada se ha generado un detector de ojos el cual nos da como resultado un archivo con la cascada de clasificadores entrenada, la cual será utilizada por nuestro sistema antes de realizar el reconocimiento. Además se han realizado una serie de pruebas con 15 diferentes sujetos, cada sujeto presenta varias sesiones de imágenes con distintas condiciones de iluminación, cercanía, etc. Se definieron distintos umbrales de certeza, lo que quiere decir que las imágenes de los sujetos similares en un valor mayor al umbral se reconoce el sujeto, en el caso de la red neuronal la salida contiene igual número de neuronas que sujetos en la base de datos, los parámetros definidos se presentan a continuación:

1. Una red neuronal del tipo: Feedforward, resilient backpropagation.
2. Una capa de entrada, una oculta y una de salida, en total 3 capas.
3. Las neuronas de entrada es igual al número de eigenvalues obtenidos por cada imagen.
4. Se definieron 10 neuronas en la capa oculta.
5. Se obtendrá una neurona en la capa de salida por cada sujeto que se encuentre en la base de datos.
6. Se define un umbral de reconocimiento del 85 %.

En relación con la implementación del algoritmo Hausdorff se estableció un umbral del 90 %, su resultado esta ligado a la respuesta de la red neuronal, en otras palabras, este algoritmo de distancia Hausdorff solo se ejecutará si el proceso de la red neuronal no supera su respectivo umbral. Finalmente solo será reconocido por la distancia Hausdorff un sujeto con el cual la red neuronal y la distancia Hausdorff coincidan en su respuesta y cuando la red neuronal no supere el valor del umbral establecido, y la distancia Hausdorff supere el suyo.

La solución propuesta se ha desarrollado con el lenguaje de programación C# utilizando Visual Studio 2012, es una aplicación de *Windows Form* que incorpora 5 proyectos de clases dentro de las cuales se encuentran la implementación de los algoritmos utilizados, se utiliza la librería EmguCV, la cual es un envoltorio para utilizar OpenCV en C# y Encog que es un framework que contiene diversos métodos de entrenamiento y generación de redes neuronales, entre las que

permite generar redes neuronales feedforward y el entrenamiento de las mismas se puede realizar mediante backpropagation [4].

### 3.1. Funcionamiento del Sistema

La solución propuesta se ha diseñado con el propósito de mostrar de una manera más clara los pasos que se realizan antes del reconocimiento.

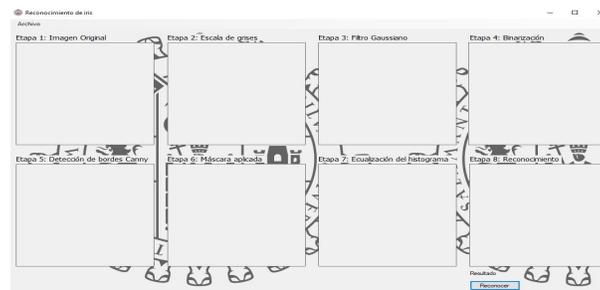


Figura 13: Pantalla del sistema

Desde la pantalla del sistema que se observa en la Fig. 13 se puede observar en la esquina superior izquierda un botón llamado Archivo, en este punto se despliega una ventana del explorador de archivos de Windows para realizar la carga de la imagen del sujeto a reconocer. Una vez cargada la imagen del sujeto a reconocer se realiza la detección del ojo y el tratamiento de la imagen, como ilustra la Fig. 14.



Figura 14: Imagen cargada en el sistema

Luego procedemos a seleccionar el botón reconocer en el cual se encuentra toda la lógica del sistema, de esta manera se pone en practica todas las teorías, algoritmos y técnicas como ser la obtención de los *eigenvalues* por parte del

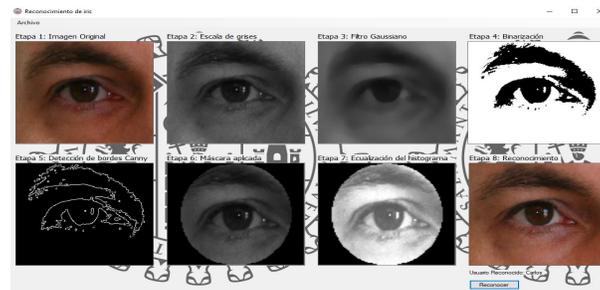


Figura 15: Resultado del sistema

algoritmo ACP, el tratamiento de la imagen que pasa muchos filtros, el potencial de las redes neuronales y la combinación con la distancia de Hausdorff.

Una vez se realiza el reconocimiento, el sistema presenta los últimos pasos realizados y presenta la pantalla final junto a una imagen del sujeto reconocido y su nombre tal y como muestra la Fig. 15. Se realizó una interfaz sencilla con la cual se puede observar de manera transparente todo el proceso que se lleva a cabo para realizar el reconocimiento del iris de un ojo.

**Experimentos del Sistema.** Para validar el desarrollo, se realizó una serie de pruebas sobre las imágenes de la base de datos UBIRIS, las cuales no fueron utilizadas en la fase de entrenamiento de la red neuronal, como mencionábamos en un apartado anterior la división de la base de datos se realizó un 80 % de ella para el entrenamiento y un 20 % para realizar las pruebas correspondientes.

Las pruebas se realizaron sobre 15 sujetos seleccionados, estos presentaban varias sesiones de imágenes que se utilizaron para probar el reconocimiento. De cada sujeto se seleccionaron varias imágenes para probar el reconocimiento y se utilizaron imágenes negativas o que no representaban el objeto a reconocer.

Sobre un total de 150 imágenes el sistema de reconocimiento pudo reconocer de manera correcta 93 imágenes que logro decir a quien pertenecían y que en realidad pertenecían a ese sujeto, 5 imágenes fueron reconocidas y las estableció como que pertenecían a un sujeto pero en realidad no pertenecían a dicho sujeto, 45 imágenes fueron catalogadas como no reconocidas y realmente eran imágenes que representaban cualquier otro objeto menos el que se pretendían reconocer, y por último 7 imágenes no fueron reconocidas pero en realidad pertenecían a los sujetos de prueba.

En total 138 de las 150 imágenes utilizadas en las pruebas, fueron reconocidas de manera correcta lo que representa un porcentaje de acierto del 92 %, para las 12 imágenes restantes que representan el 8 %, el reconocimiento fue realizado de manera errónea. La Tab. 4 muestra la matriz de confusión obtenida por el sistema.

Cabe resaltar que de los experimentos realizados, se omitieron las imágenes que presentaban oclusión parcial del iris del ojo como ser cabello que ocultaba

Clase real Total		
TP	TF	
93	5	98
FN	TN	
7	45	52

Tabla 1: Matriz de confusión.

parte del iris, gafas oscuras o lentes, también las imágenes en las cuales el ojo se encontraba en ángulos donde no se observaba correctamente el iris; ya que al utilizar este tipo de imágenes los resultados que nos generaba el sistema se verían afectados. Por lo que se utilizaron solo imágenes del ojo de alta resolución y en condiciones de laboratorio a una distancia no menor a 50 cm, se presenta en el apartado de líneas de trabajo futuro como un factor importante a tomar en cuenta para este tipo de reconocimiento.

## 4. Conclusiones y Trabajo Futuro

En la presente investigación, se estudiaron distintos algoritmos para el problema de reconocimiento del iris. A continuación se presentan las conclusiones a las que se llegó y las líneas de trabajo futuro.

### 4.1. Conclusiones

La propuesta es una combinación de algunas técnicas implementadas en otros problemas de reconocimiento; la solución propuesta presenta un enfoque para la detección el cual se basa en el algoritmo de detección de Viola y Jones[7] que consiste en entrenar una cascada de clasificadores, en este método se entrena el detector utilizando un algoritmo de Adaboost y luego se utiliza la cascada de clasificadores resultante para la detección. Además presenta un segundo enfoque para el reconocimiento, el cual se basa en la extracción de características de una imagen utilizando el algoritmo ACP y de esta manera se obtiene un arreglo de eigenvalues que servirán de entrada de una red neuronal que realiza el reconocimiento utilizando una función sigmoidea como salida; por último para darle más robustez al proyecto se realiza una implementación de la medida de distancia de Hausdorff que se aplica sobre la conversión en mapas de bordes de la imagen, que frente a los problemas de iluminación, que son un ruido común en las imágenes del iris nos proporciona un mejor rendimiento.

El algoritmo de análisis de componentes principales (ACP), es una teoría muy extendida y utilizada en el reconocimiento facial, pero su simplicidad, velocidad y eficacia a la hora de la extracción de características puede ser aplicada fácilmente en otros problemas de reconocimiento como el presentado en este trabajo. La combinación de ACP junto con las redes neuronales y la utilización de la distancia de Hausdorff nos da como resultado una solución bastante eficaz para el

reconocimiento del iris.

La solución muestra la vulnerabilidad que puede tener el algoritmo de análisis de componentes principales (ACP) especialmente cuando se presenta un fondo en la imagen, por lo que se aplica una máscara sobre ésta para retirar el fondo y que no interfiera en el reconocimiento. Los cambios de iluminación no constituyen un mayor problema; sin embargo, finalmente un sistema para que sea realmente robusto y confiable debe de considerar muchos factores que abren diversas líneas de investigación futura.

#### 4.2. Líneas de Trabajo Futuro

Basados en los resultados y las conclusiones que se han presentado, podemos extraer algunas líneas de trabajo futuro que se mencionan a continuación:

- Como ser el diseño de otros algoritmos que mejoren los resultados bajo condiciones de oclusión, así de esta manera trabajar para que las características ocluidas no se mezclara con las válidas.
- Además de la consideración del fondo de las imágenes, tomar en cuenta la presencia de detalles cerca del ojo como ser cabello, gafas, lentes, etc.
- Sería interesante estudiar técnicas de alineamiento del ojo, ya que ésta es una de las causas mas importantes de la baja precisión en los sistemas de reconocimiento del iris, se debe a problemas que tengan que ver con la orientación del ojo.
- Estudiar las técnicas para la detección del ángulo de rotación y conversión de la imagen para mejorar el reconocimiento en estos casos.
- La implementación de redes neuronales convolucionales, ya que estas abordan de mejor manera los problemas con imágenes y se han demostrado sus buenos resultados en otros sistemas de reconocimiento por lo cual pueden ayudar a mejorar estos sistemas.

## Referencias

1. "Open source computer vision library, cascade classifier training." [Online]. Available: [http://docs.opencv.org/trunk/dc/d88/tutorial\\_traincascade.html](http://docs.opencv.org/trunk/dc/d88/tutorial_traincascade.html)
2. "Open source computer vision library, tutorials." [Online]. Available: <http://docs.opencv.org/2.4/doc/tutorials/tutorials.html>
3. L. Alonso Romero and T. Calonge Cano, "Redes neuronales y reconocimiento de patrones," 2001.
4. J. Heaton, "Encog: library of interchangeable machine learning models for java and c#." *Journal of Machine Learning Research*, vol. 16, pp. 1243–1247, 2015.
5. D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the hausdorff distance," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 15, no. 9, pp. 850–863, 1993.
6. L. I. Smith, "A tutorial on principal components analysis," Tech. Rep., 2002.
7. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–I.

# Sistema de Gestión de Consumo Adaptativo para la Gestión de la Demanda en Redes Eléctricas

Diego M. Jiménez-Bravo<sup>1</sup>, Juan M. Corchado<sup>1,2,3</sup>,  
Juan F. De Paz<sup>1</sup> y Gabriel Villarrubia González<sup>1</sup>

<sup>1</sup> BISITE Digital Innovation Hub, Universidad de Salamanca,  
Edificio Multiusos I+D+I, 37007, Salamanca, España  
{dmjimenez, corchado, fcofds, gvg}@usal.es  
<http://bisite.usal.es/>

<sup>2</sup> Osaka Institute of Technology, Japan

<sup>3</sup> Universiti Malaysia Kelanta, Malaysia

**Resumen** El control del consumo en las viviendas y edificios de trabajo es un aspecto cada vez más importante, debido principalmente a la aparición de las ciudades inteligentes, el aumento del consumo de energías renovables y las políticas de la Unión Europea por el uso de la energía de una forma eficiente y limpia. Es por ello que es de vital importancia la realización de un sistema que permita predecir que dispositivos se encuentran conectados a una red eléctrica y que permita actuar sobre ellos en caso de que fuese necesario. En esta memoria se plantea la utilización de un sistema multiagente que introduce sistemas de razonamiento basado en casos para cumplir dichos objetivos. Para ellos se realizara el sistema utilizando diferentes técnicas de *machine learning* y comparando cual de ellas nos ofrecen mejores resultados de cara a realizar una predicción.

**Keywords:** predicción eléctrica, NILM, minería de datos, *machine learning*, sistema multiagente, razonamiento basado en casos.

## 1. Introducción

Tal y como relatan De Baest et al. [1] en su artículo, en octubre de 2014 los países miembros de la unión europea se reunieron para acordar una serie de objetivos de cara al año 2030. Estos objetivos marcados hacían referencia a la reducción de contaminación y obtener una energía más eficiente y limpia. Es por ello, que se plantea el aumento del uso de energías renovables de cara a los próximos años. Este tipo de energías suponen un desafío desde el punto de vista de la eficiencia, ya que es mucho más difícil ajustar la producción a la demanda. De esta forma, cobra vital importancia la gestión del consumo en las líneas eléctricas.

El control de consumo de las redes es un aspecto que cobra cada vez más relevancia a parir de la aparición de las *Smart Grids*. Las fuentes de producción son cada vez más diversas y se encuentran distribuidas geográficamente por lo

que es más complicado mantener la red eléctrica en condiciones estables. Por este motivo, cobra interés la realización de sistemas que permitan gestionar el consumo de manera automática para ajustarse a la producción generada en cada instante puesto que la producción a corto plazo no es factible gestionarla.

A lo largo de los últimos años se han empezado a confeccionar y a aparecer cada vez más dispositivos que permiten realizar un análisis del consumo eléctrico en las viviendas y edificios de trabajo. En la era de la información estos datos son de un alto valor, ya que permiten extraer conocimiento de ellos. Por ejemplo, se puede obtener los datos de consumo de una vivienda y de esta forma poder analizar la información de consumo para detectar patrones de comportamiento y analizar los dispositivos conectados a esa red doméstica.

Por lo tanto, se plantea la necesidad de realizar un sistema multiagente que sea capaz de analizar los datos procedentes de algún tipo de sistema de medición que sea capaz de proporcionar los datos necesarios. El análisis de los datos de consumo por lo general tiene diferentes fases, extracción de los datos, preprocesamiento de los datos, clasificación y extracción de conocimiento para actuar sobre los dispositivos. Para realizar estos sistemas, sobre todo el de clasificación, existen diferentes algoritmos que nos pueden permitir realizar la clasificación e identificación de los diferentes dispositivos conectados a la red eléctrica.

Es por ello, que se hace necesario integrar un sistema de razonamiento basado en casos junto con la arquitectura del sistema multiagente. Los sistemas de razonamiento basados en casos se basan en el conocimiento pasado, por lo que esta característica se acopla perfectamente a la naturaleza de esta investigación.

De aquí en adelante el contenido de este artículo contará con las siguientes secciones, la sección 2 donde se abordarán los estudios realizados hasta la fecha en este campo de la investigación. La sección 3 donde se planteará el sistema propuesto. La sección 4 de resultados, donde se expondrá el rendimiento que obtiene el sistema planteado. Y por último, en la sección 5 de conclusiones donde se recogerán las opciones que nos permite realizar el sistema planteado así como posibles mejoras o trabajos que se puedan realizar de este estudio.

## 2. Estado del arte

En la literatura existen una gran variedad de trabajos relacionados con la predicción de dispositivos conectados a una red eléctrica y/o predicción del consumo a corto plazo. Para ello esos trabajos utilizan diferentes técnicas pero todos ellos se basan en la técnica de *nonintrusive load monitoring* (Hart [11] y Zoha et al. [16]). En esta sección se mostraran y analizarán brevemente algunos de estos trabajos.

En el trabajo propuesto por De Baesta et al. [1] se propone dos nuevos y diferentes algoritmo con el objetivo de poder realizar la técnica NILM (*NonIntrusive Load Monitoring*) y de esta forma poder saber que dispositivos se encuentran conectados en la red. Proponen dos nuevos métodos, una versión del *chi* cuadrado con bondad de ajuste y una detención de eventos basados en el suavizado de *cepstrum*. Además, establecen que ambos métodos pueden ser optimizados mediante la técnica de sustitución.

Otros estudios científicos proponen una mejora a las técnicas NILM. Tang et al. [14] proponen tener en cuenta cuando la vivienda o establecimiento del que se obtienen los datos se encuentra ocupado o no. De tal forma que cuando la casa se encuentra desocupada no realizará ninguna acción y cuando la vivienda se encuentre ocupada aplicará las técnicas de desagregación necesarias para obtener los electrodomésticos que se encuentran conectados. Esta técnica según los autores reduce la computación al no analizar los periodos donde los edificios están desocupados y aumenta la precisión en la clasificación de los dispositivos conectados a la red eléctrica.

El estudio realizado por Brown et al. [4] también propone tener en cuenta cuando un edificio se encuentra ocupado o no. Para detectar cuando un edificio o vivienda se encuentra ocupado utilizan un radar de banda ancha comparando los resultados de este con los datos recibidos del contador. Utilizando estas dos señales se podría saber el estado del edificio y los dispositivos electrónicos conectados en el mismo. Además, con esto se podrían extraer patrones de comportamiento de los usuarios.

Por el contrario, Lin et al. [12] ofrecen una nueva propuesta para los sistemas basados en NILM. Proponen identificar los dispositivos basándose en los principios de la programación cuadrática. Las pruebas realizadas con el modelo demuestran que con esta metodología se consiguen resultados más que apropiados.

Existen otros tipos de estudios que lo que pretenden es predecir la demanda eléctrica en los días futuros. Existen diferentes artículos tratando este tema como por ejemplo los propuestos por Chen et al. [7] y por Chen y Tan [6]. El primero de ellos pretende obtener la demanda eléctrica horaria por día, para ello utiliza un algoritmo de *clustering* híbrido. El segundo realiza la misma acción con un algoritmo híbrido de regresión del vector de soporte. Ambos trabajos ofrecen buenos resultados de cara a la predicción horaria de la demanda energética.

Por último, merece la pena mencionar que existen herramientas que permiten realizar la desagregación mediante técnicas NILM. Como por ejemplo la herramienta de código abierto *NILMTK* (*NonIntrusive Load Monitoring ToolKit*) desarrollada por Batra et al. [2]. Permite comparar diferentes técnicas NILM mostrando los resultados que ofrecen cada una de ellas sobre los diferentes *datasets* que introduce la herramienta.

### 3. Propuesta

El sistema de gestión de consumo adaptativo para la gestión de la demanda en redes eléctricas pretende ser capaz de comunicarse con diferentes medidores de tal manera que sea capaz de leer la información que estos gestionan y que sea capaz de actuar sobre ellos. El objetivo de leer la información que gestionan es que el sistema sea capaz de poder tratar y extraer la cantidad de información necesaria para poder identificar los electrodomésticos conectados a una red eléctrica.

Para poder alcanzar el objetivo, el sistema se organiza en una serie de agentes. Los agentes son los encargados de realizar cada una de los subprocesos que tiene

el sistema global. Los agentes que forman parte del sistema se pueden dividir en dos grandes grupos, tal y como se ve en la figura 1. Por un lado se tendrían los agentes de entorno y por otro los agentes del procesamiento.

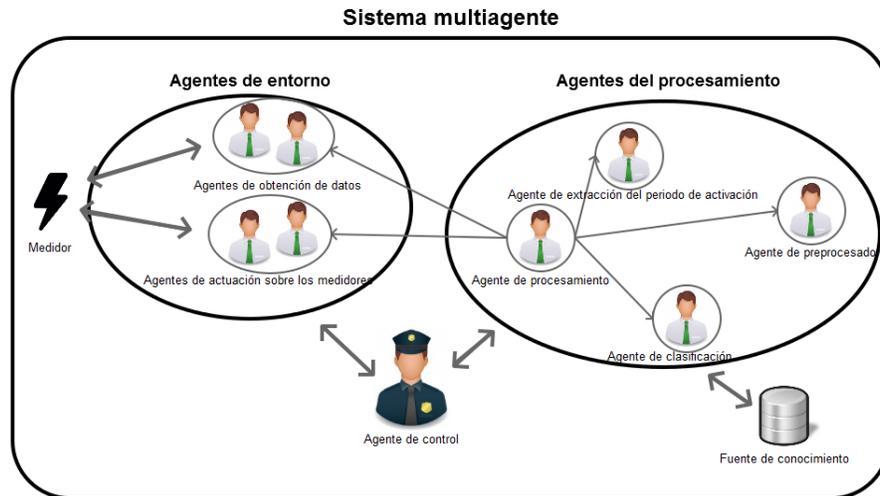
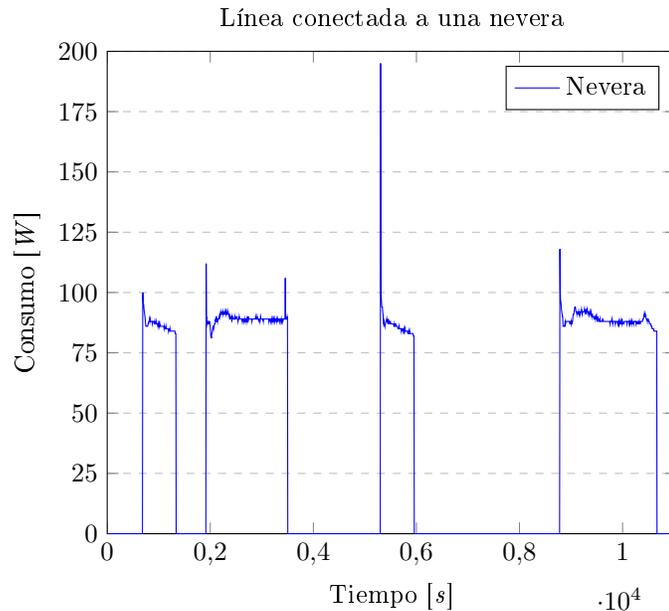


Figura 1: Arquitectura del sistema multiagente planteado.

- **Agentes de entorno.** Son los encargados de comunicarse con los sistemas de medición que se incorporen en el sistema. Se encargarán tanto de la adquisición de los datos como de actuar sobre el sistema de medición. Los agentes de obtención de datos se comunicarán con los medidores mediante los protocolos requeridos para obtener los datos. Mientras que los agentes de actuación sobre los medidores serán los encargados de encender o apagar las diferentes líneas a las que los medidores se encuentren conectadas.
- **Agentes de procesamiento.** Se encargan de realizar todo el procesamiento interno de la información así como de preprocesarla y demás acciones. Son los encargados de realizar las principales tareas del sistema. El agente de procesamiento es el encargado de gestionar al resto de agentes de este grupo. El agente de extracción de periodos de activación se encarga de extraer los periodos de activación de los datos en bruto que recibe.



De lo que se encargaría este agente sería de extraer los valores de consumo de cada periodo de activación, de tal forma que se tengan agrupados los valores de consumo por cada uno de los periodos de activación. De tal forma que del gráfico anterior extraería cuatro periodos de activación. El agente de preprocesado es el encargado de extraer la información necesaria e importante para el proceso de identificación de los periodos de activación identificados por el agente descrito con anterioridad. Las características que se han de extraer se detallan a continuación.

- Media. Esta característica indica la media de los valores de consumo por cada periodo de activación, es decir es la media aritmética del consumo que tiene cada dispositivo en sus periodos de activación.
- Máximo. Indica el valor máximo de consumo que ha tenido el dispositivo que se está evaluando durante el periodo de activación recibido.
- Mínimo. Indica el valor mínimo de consumo que se encuentra en el periodo de activación que es objeto de estudio.
- Primer escalón. Este valor indica la diferencia que existe con el primer valor del periodo de activación con respecto al valor máximo. De tal forma que se tenga en cuenta el “salto” que se produce.
- Segundo escalón. En este caso, el valor representa la diferencia que existe entre el máximo valor de consumo del periodo con respecto al último valor del periodo. De tal forma que se tenga en cuenta también este segundo “salto” que se produce en la expresión del dispositivo.
- Tiempo. Este campo indicará el tiempo en minutos en el que el dispositivo ha estado activo, es decir, el periodo de activación del aparato que se está tratando.

El agente de clasificación es el encargado de determinar que tipo de electrodoméstico es el que se está tratando. Es decir, el agente es capaz de predecir que tipo de dispositivo es al que pertenecen las características que ha extraído el anterior agente. En este caso para determinar de que electrodoméstico se trata este agente incorpora un sistema de razonamiento basado en casos. Para el planteamiento de este sistema se plantean utilizar diferentes algoritmos dentro de este agente (algunos de ellos se definen en [10], [3], [5], [15], [8]), con el objetivo de determinar que tipo de técnicas nos ofrecen unos mejores resultados tal y como se expondrá en la sección 4.

- Agente de control. Es el encargado de controlar todos los agentes del sistema que se encuentran distribuidos en alguno de los grupos anteriores.

Los agentes de los diferentes grupos se intercomunican entre sí para poder lograr el objetivo final del sistema. De esta forma en la figura 1 se muestran las conexiones que existen entre los diferentes agentes del sistema.

## 4. Resultados

En esta sección se explica y analizan los resultados obtenidos de las pruebas que se llevan a cabo. Como se ha mencionado antes el sistema a de ser capaz de identificar diferentes dispositivos. Para ello se han utilizado diferentes algoritmos de *machine learning* para así poder comparar y determinar cual o cuales de ellos ofrecen mejores resultados. Los cuales son los siguientes, el algoritmo RIPPER, el PART, el C4.5, el *RandomForest*, el *RandomTree*, el *REPTree*, el *k-NN*, el *KStart*, redes bayesianas y redes neuronales. Las versiones de los algoritmos que se utilizan para el sistema que se plantea son las versiones que trae por defecto la librería de *machine learning Weka*.

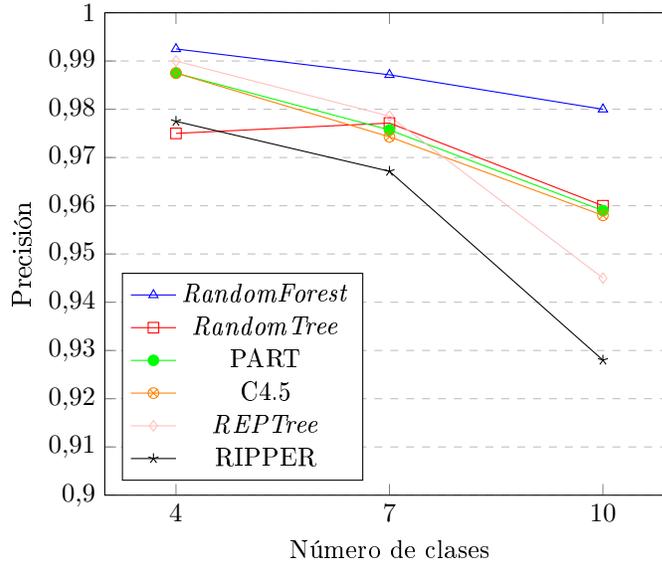
Los resultados obtenidos se obtienen con un *dataset* que contiene diez electrodomésticos, los cuales son una nevera, una bomba de agua, una televisión, un lavavajillas, una lavadora, una tetera, un calentador de gas, un congelador, un microondas y una impresora led. Compuesto por un total de mil instancias, cien por cada una de las clases.

Dado que los algoritmos y métodos de clasificación que mejor funciona son los formados por reglas de decisión y árboles, se puede decir que estos algoritmos son los que mejores resultados ofrecen de cara a predecir que electrodomésticos se encuentran conectados a una red eléctrica. Por ello, se muestra en el gráfico 4 una comparativa de este tipo de clasificadores en cada una de las pruebas que se ha realizado.

Algoritmo	Precisión	kappa
RIPPER	0,928	0,92
PART	0,959	0,9544
C4.5	0,958	0,9533
<i>RandomForest</i>	0,98	0,9778
<i>RandomTree</i>	0,96	0,9556
<i>REPTree</i>	0,945	0,9389
<i>k-NN</i>	0,952	0,9467
<i>kStart</i>	0,966	0,9622
Redes bayesianas	0,927	0,9189
Redes neuronales	0,853	0,8367

Tabla 1: Resultados de los algoritmos con el *dataset* de diez clases.

Resultados de los clasificadores basados en reglas en los diferentes tests

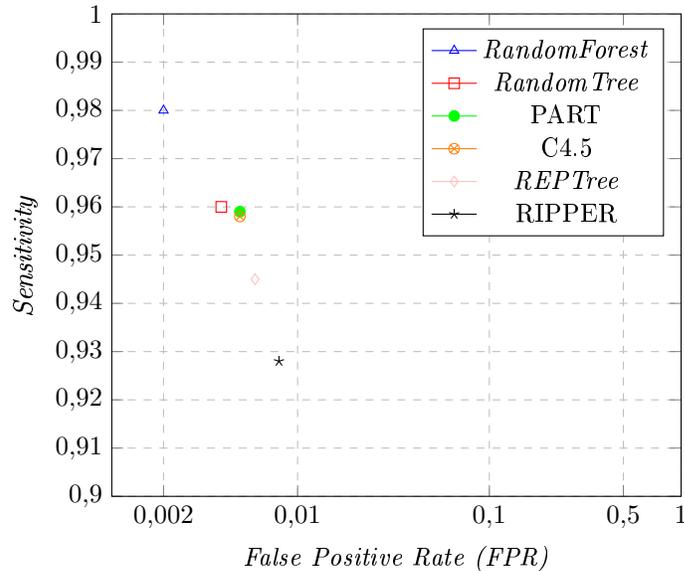


Como se puede observar todos los algoritmos basados en reglas siguen un comportamiento similar. A medida que se van añadiendo nuevas instancias al conjunto de datos su precisión se ve disminuida. A excepción del *RandomTree* que en el la segunda prueba obtiene una ligera mejoría. Pero por lo general, todos los clasificadores que se han probado obtienen buenos resultados, lo que supondría que prácticamente cualquiera de ellos podría servir para la predicción de dispositivos en una red. No obstante, el que mejores resultados ofrece es el *RandomForest*.

Esto se puede comprobar mediante el area bajo la curva ROC o mediante la distancia del punto formado por los valores (*false positive rate*, *sensitivity*) de cada clasificador al punto (0, 1) tan y como se explica Fawcett [9]. Por ello en la

gráfica 4 se muestra la posición de esos puntos, de los clasificadores de la gráfica anterior, en un espacio bidimensional.

Posición de los puntos (*false positive rate*, *sensitivity*) de los clasificadores

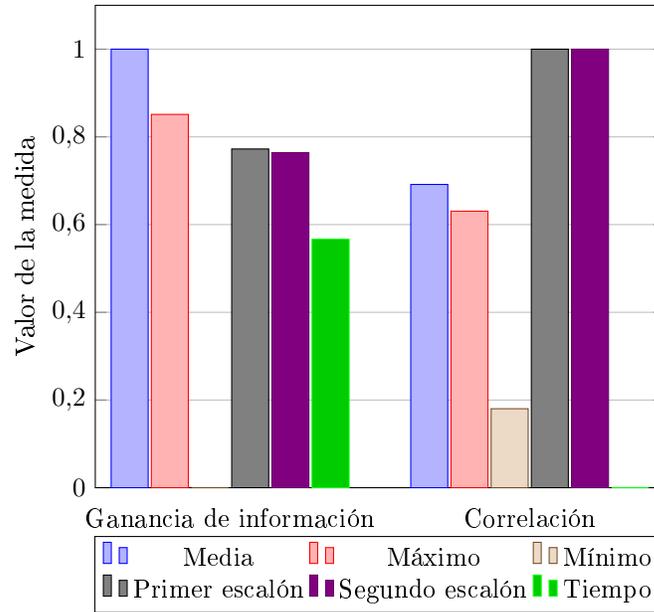


Como se puede observar los algoritmos que a primera vista se encuentran más cercanos al punto (0, 1) son los que en el resto de gráficas y tablas ofrecen también unos mejores resultados.

Pero sin duda alguna, el que todos los algoritmos funcionen tan bien se debe a la composición del *dataset*. Y es que la selección de los atributos de los que se compone el conjunto de datos es fundamental para que cada uno de los métodos de *machine learning* ofrezca los resultados mostrados en las tablas y la gráfica anterior. Por ello, sería interesante realizar una selección de atributos con diferentes técnicas para comprobar que atributos son los que aportan una mayor información a la hora de predecir la clase.

Para realizar esta evaluación se utilizarán dos técnicas diferentes la ganancia de información y la correlación de cada uno de los atributos con respecto a la clase. En la gráfica 4 se muestran una comparativa de las dos técnicas utilizadas, donde los resultados de la tabla han sido normalizados, mediante la fórmula del escalado de variables, con el objetivo de que la comparación sea más clara y objetiva.

Resultados de las selecciones de atributos con los resultados normalizados



Se puede observar en la gráfica 4, que en las dos técnicas de selección de atributos, las características que por lo general ofrecen un mayor información a la hora de predecir la clase son la media, el valor máximo y el primer y segundo escalón. Se observa que tanto el mínimo como el tiempo no ofrecen tanta información a la hora de realizar la clasificación. Estas características son las que permiten que en muchos de estos métodos de clasificación el éxito a la hora de clasificar sea tan alto.

## 5. Conclusiones

Durante la elaboración de este trabajo se ha presentado un sistema que es capaz de determinar que dispositivos se encuentran conectados a una red eléctrica con el objetivo de poder gestionarlos de tal forma que si es necesario desconectar alguno de ellos de la red dado que la producción de energía en ese momento es escasa o la red está teniendo un consumo excesivo, se puedan desconectar alguno de estos dispositivos que no son necesarios o fundamentales en las viviendas, oficinas, etc.

El sistema planteado es un sistema multiagente donde se tienen diferentes tipos de agentes cada uno de los cuales se encarga de una función diferente en el proceso. El sistema de agentes incluye también el uso de sistemas de razonamiento basado en casos.

El trabajo deja la puerta abierta a una gran cantidad de posibles mejoras y ampliaciones. Añadir más cantidad de dispositivos y electrodomésticos al sis-

tema. De esta forma, se podrían identificar nuevos dispositivos y hacer así al sistema más completo. Añadir nuevos electrodomésticos al sistema hará que el mismo sea más robusto a la hora de ser instalado en una vivienda por ejemplo, pues cuanto más se acople la base de datos de dispositivos a los de una casa real más eficaz será el sistema. Si se introducen nuevos dispositivos al sistema, sería interesante comprobar de nuevo como se comportan los algoritmos estudiados y dependiendo de funcionamiento de los mismos, quizás sea necesario estudiar la aplicación de otros algoritmos para que se puedan mejorar los resultados. Se podría añadir una nueva técnica a la hora de extraer el periodo de activación de cada uno de los dispositivos conectados a la red. Existen técnicas que permitirían sacar el periodo de activación, como la propuesta por Serrà y Arcos [13]. Otra aspecto sin duda a tener en cuenta es la seguridad. Se podría dotar de seguridad a las comunicaciones que se establecen con los medidores a los que está conectado el sistema. El motivo de realizar esta mejora, es que la información que se transmite pueden considerarse importantes y debe garantizarse la privacidad de los datos de los usuarios del sistema inteligente planteado.

**Agradecimientos** Este trabajo ha sido financiado por la European Commission H2020 MSCA-RISE-2014: Marie Skłodowska-Curie project DREAM-GO Enabling Demand Response for short and realtime Efficient And Market based smart Grid Operation—an intelligent and real-time simulation approach ref 641794.

## Referencias

1. L. D. Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "On the bayesian optimization and robustness of event detection methods in {NILM}"," *Energy and Buildings*, vol. 145, pp. 57 – 66, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S037877881631605X>
2. N. Batra, J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, A. Singh, and M. Srivastava, "Nilmtk: an open source toolkit for non-intrusive load monitoring," in *Proceedings of the 5th international conference on Future energy systems*. ACM, 2014, pp. 265–276.
3. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. [Online]. Available: <http://dx.doi.org/10.1023/A:1010933404324>
4. R. Brown, N. Ghavami, H.-U.-R. Siddiqui, M. Adjrard, M. Ghavami, and S. Dudley, "Occupancy based household energy disaggregation using ultra wideband radar and electrical signature profiles," *Energy and Buildings*, vol. 141, pp. 134 – 141, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0378778816317261>
5. L. D. C, "Article: Comparative analysis of random forest, rep tree and j48 classifiers for credit risk prediction," *IJCA Proceedings on International Conference on Communication, Computing and Information Technology*, vol. ICCCMIT 2014, no. 3, pp. 30–36, March 2015, full text available.
6. Y. Chen and H. Tan, "Short-term prediction of electric demand in building sector via hybrid support vector regression," *Applied Energy*, 2017.

7. Y. Chen, H. Tan, and U. Berardi, "Day-ahead prediction of hourly electric demand in non-stationary operated commercial buildings: A clustering-based hybrid approach," *Energy and Buildings*, vol. 148, pp. 228 – 237, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0378778816313792>
8. J. G. Cleary and L. E. Trigg, "K\*: An instance-based learner using an entropic distance measure," in *In Proceedings of the 12th International Conference on Machine Learning*. Morgan Kaufmann, 1995, pp. 108–114.
9. T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.
10. E. Frank and I. H. Witten, "Generating accurate rule sets without global optimization," in *Proceedings of the Fifteenth International Conference on Machine Learning*, ser. ICML '98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 144–151. [Online]. Available: <http://dl.acm.org/citation.cfm?id=645527.657305>
11. G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the Ieee*, vol. 80, no. 12, pp. 1870–1891, 1992. [Online]. Available: <GotoISI>://WOS:A1992KL37100002
12. S. Lin, L. Zhao, F. Li, Q. Liu, D. Li, and Y. Fu, "A nonintrusive load identification method for residential applications based on quadratic programming," *Electric Power Systems Research*, vol. 133, pp. 241 – 248, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0378779615003910>
13. J. Serrà and J. L. Arcos, "Particle swarm optimization for time series motif discovery," *Knowledge-Based Systems*, vol. 92, pp. 127–137, 2016. [Online]. Available: <http://authors.elsevier.com/a/1SBMd3OAb8gdos>
14. G. Tang, Z. Ling, F. Li, D. Tang, and J. Tang, "Occupancy-aided energy disaggregation," *Computer Networks*, vol. 117, pp. 42 – 51, 2017, cyber-physical systems and context-aware sensing and computing. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1389128616304133>
15. I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., 2011.
16. A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16 838–16 866, 2012. [Online]. Available: <GotoISI>://WOS:000312607500047

# Sistema Inteligente para la Generación de Armonías

Lucía Martín-Gómez, Belén Pérez Lancho y María Navarro-Cáceres

BISITE Digital Innovation Hub, University of Salamanca. Edificio Multiusos I+D+i,  
37007, Salamanca, Spain

luciamg@usal.es, lancho@usal.es, maria90@usal.es

**Resumen** Las imágenes y la música son dos formas de expresión cultural muy comunes en la actualidad. El componente emocional de ambas es la base para la inspiración de este trabajo, siendo la sinestesia el fenómeno utilizado como vínculo de unión. La propuesta define un sistema de creación musical automática inspirada en los colores de una imagen. Para ello se hace uso del algoritmo *Particle Swarm Optimization*, con el que se obtiene una progresión de acordes con una buena calidad armónica. El sistema se ha evaluado mediante dos tests de usuarios, el primero para validar el procedimiento de traducción color-sonido utilizado y el segundo para comprobar la calidad de los resultados musicales obtenidos.

**Keywords:** Música, color, sinestesia, Inteligencia Artificial, *Swarm Intelligence*, *Particle Swarm Optimization*

## 1. Introducción

A lo largo de la historia la música ha sido uno de los pilares básicos de la cultura, influyendo y condicionando las costumbres de un pueblo y las emociones de las personas. Por otra parte, históricamente, las imágenes son uno de los medios de comunicación más relevantes. Actualmente, los diversos avances de la tecnología hacen posible la obtención de imágenes de todo lo que nos rodea de manera rápida y sencilla.

La música y las imágenes comparten un componente sentimental y emocional. Basándose en ello, uno de los vínculos que se han establecido entre las imágenes y la música a lo largo de la historia es la sinestesia, que es un fenómeno neurológico que conjuga dos fenómenos perceptuales o estímulos. Además, ha sido utilizado como inspiración para la creación de obras que integran diferentes campos perceptivos; es decir, obras que contienen información capaz de estimular varios sentidos de manera simultánea. La relación más utilizada en este campo es la que une color y sonido. Esta asociación ha dado lugar a creaciones artísticas de diferentes ramas como la pintura, la música o la literatura. Aristóteles estudiaba las armonías del color y sonido [29], Isaac Newton propuso que las notas musicales y los colores compartían características comunes [11] y Johann Wolfgang von Goethe decía que es necesario contemplar las artes en correspondencia unas con otras, describiendo en [44] cómo son percibidos por los humanos. Al escuchar o

componer música, en la cabeza de Olivier Messiaen se reproducían secuencias de colores. Este tipo de experiencias dio lugar a las obras [34] y [35].

Todas estas obras respaldan que la sinestesia puede considerarse como un componente esencial del arte que permite ir más allá en el proceso de creación convencional aunando el sonido, el color y el sentimiento.

En este trabajo se aplica el algoritmo evolutivo *Particle Swarm Optimization* (PSO) para obtener un fragmento musical armónico inspirado en las características de una imagen determinada. Como punto de partida, los colores de la imagen serán extraídos y traducidos a sonidos con ayuda de un sistema basado en sinestesia. Estos sonidos serán reagrupados en base a las reglas de la armonía clásica [31]. Adicionalmente, para lograr una mejor comprensión del proceso de composición se utilizarán técnicas de visualización y de sonificación.

La Sec. 2 hace un repaso sobre las distintas técnicas de composición musical automática, el tratamiento del color y del sonido y algunas técnicas de visualización recogidas en la literatura. En la Sec. 3 se describe un trabajo anterior en el que planteábamos el uso de una imagen para la generación de una composición musical sencilla. El grueso de la propuesta de este trabajo se describe en la Sec. 4. En la Sec. 5 se discute sobre la calidad de las composiciones obtenidas. Finalmente, en 5 se presentan las conclusiones finales de este trabajo y se proponen las líneas de trabajo futuro.

## 2. Estado del Arte

En esta sección se repasan los principales enfoques que han tenido en la literatura los problemas de extracción de color, traducción de este a sonido, representación matemática del sonido y algunas propiedades musicales y las técnicas de composición automática. Adicionalmente, se realizará un estudio de algunas técnicas de visualización que podrían aplicarse para involucrar al usuario en el proceso creativo [38].

### 2.1. Tratamiento del Color

El punto de partida del proceso creativo en este trabajo es una imagen, cuyos colores serán utilizados como fuente de inspiración para la composición musical.

Para poder llevar a cabo el sistema de composición propuesto es necesaria la extracción de colores de la imagen de partida. Se han analizado algunas técnicas relativas a la extracción del color en imágenes, tales como los histogramas de color y la cuantificación.

Debido a la ingente cantidad de colores que presenta una imagen en formato digital, se considera que, en este problema, la aplicación de la teoría de cuantificación es muy apropiado por su capacidad de reducción de información de un rango de valores a un valor único. Entre los diferentes algoritmos se ha elegido el denominado *Median Cut* debido a su robustez y a la obtención directa de los  $n$  colores más representativos. Con él se subdivide el espacio del color en cajas rectangulares, y para cada una de ellas se almacena la relación entre su color y

los colores de las cajas adyacentes. Como resultado final se obtiene una paleta de color con aquellos más representativos [20].

Adicionalmente, para poder tratar el color de forma automática es necesario obtener una representación matemática del mismo, para lo cual se utilizan los modelos de color [21]. Entre ellos destacan el RGB, el CMYK, el YUV o el HSV, que es el elegido para este trabajo. Este modelo representa las características fundamentales del color: matiz, saturación y brillo. Está inspirado en la propuesta de Munsell [36], que constituyó la primera aproximación a una representación del color. Su característica más remarcable es que se basa en el sistema visual humano de percepción de los colores, lo que hace que el modelo sea fácilmente interpretable por el usuario. La Fig. 1 muestra una representación visual de cada uno de estos modelos.

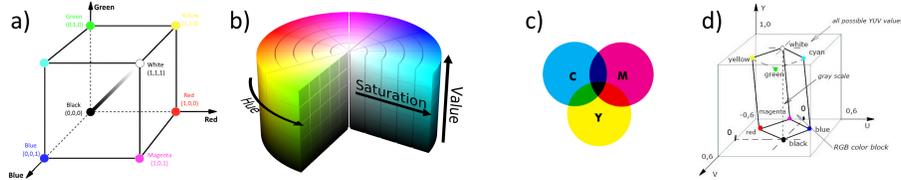


Figura 1: Modelos de color RGB, HSV, CMYK y YUV respectivamente

## 2.2. Representación de Configuraciones Sonoras

Igual que sucedía con el color, existen diversas técnicas y modelos que permiten representar y medir el sonido de manera numérica o matemática.

El formato MIDI [41] permite el almacenamiento de datos que describen la música numéricamente de manera sencilla y estandarizada. En este formato las notas musicales se representan mediante un número entero positivo comprendido entre 0 y 127, con lo cual se cubren 11 octavas musicales; sin embargo es una codificación que no resulta adecuada para la extracción de características musicales y de las relaciones entre las notas.

Algunos autores proponen una organización de las notas en espacios geométricos de manera que ciertas propiedades como la cercanía reflejen las características armónicas de las que se hablaba anteriormente. Uno de los primeros ejemplos de estas representaciones geométricas es el *Tonnetz* [10]. [9] propone un modelo tridimensional en espiral y en [19] el espacio se representa como un toroide. En [3], se extienden estos espacios para calcular las características armónicas de todas las configuraciones musicales posibles (acordes, intervalos, escalas, tonalidad...) mediante el cálculo ponderado de la Transformada Discreta de Fourier (DFT) en vectores denominados vectores de croma.

El *Tonal Interval Space* (TIS) [37] es otra propuesta de representación de configuraciones musicales basada en el uso del concepto de los vectores de croma propuestos en [3]. Estos son unos vectores dodecadimensionales binarios, donde cada posición hace referencia a una de las 12 notas de la escala cromática tal y

como se puede observar en la Tab. 1, y cada valor del vector indica la presencia (1) o ausencia (0) de dicha nota en la configuración musical deseada.

	Vector de Croma											
Posición	0	1	2	3	4	5	6	7	8	9	10	11
Nota	<i>Do</i>	<i>Do♯</i>	<i>Re</i>	<i>Re♯</i>	<i>Mi</i>	<i>Fa</i>	<i>Fa♯</i>	<i>Sol</i>	<i>Sol♯</i>	<i>La</i>	<i>La♯</i>	<i>Si</i>
ReM (acorde)	0	0	1	0	0	0	1	0	0	1	0	0

Tabla 1: Representación del vector de croma del acorde de Re Mayor

Mediante el cálculo de la Transformada Discreta de Fourier [6] de estos vectores se obtienen otros vectores denominados *Tonal Interval Vectors* (TIV), a partir de los cuales se pueden establecer medidas de la calidad musical basadas en cálculos geométricos de distancias euclídeas y angulares.

En este trabajo las notas serán representadas con el formato estandarizado MIDI. Sin embargo, debido a que la información que proporciona es insuficiente para el cálculo de las propiedades musicales necesarias para realizar la composición, se combinará con otras técnicas más complejas como son la DFT y el TIS.

### 2.3. Traducción Color - Sonido

Para realizar una composición musical basada en imágenes, es necesario establecer una relación entre dos magnitudes representativas, como son el color y el sonido. Probablemente, el primer acercamiento fue la Rueda musical del color de Newton [11], en la cual se establecía una relación entre los 7 colores del prisma con las 7 notas de una escala diatónica. Sin embargo, la existencia de 12 notas musicales en la escala cromática debilitan las bases de esta teoría.

Algunos investigadores han propuesto sistemas de traducción automática del color al sonido, dando lugar a aplicaciones como *Monalisa App* [23], que permite convertir imágenes en sonidos y viceversa mediante el tratamiento de los mismos como una secuencia de números binarios o *Harmonic Paper* [24], que toma como punto de partida un dibujo realizado por el usuario para realizar una traducción automática y obtener una composición musical. Otros autores dan un paso más allá y proponen la combinación de la sinestesia con diversas técnicas de la Inteligencia Artificial. Un ejemplo de ello podría ser [43], que compone melodías a partir de los estímulos físicos que una gota de mercurio recibe por parte de un usuario.

Por otra parte, el fenómeno denominado percepción continua [18] hace referencia a la forma gradual en que los humanos perciben los cambios. Este fenómeno podría aplicarse para relacionar los diferentes brillos de un color concreto con la intensidad del sonido, determinada por su amplitud de onda. En la Fig. 2 se muestra la relación entre las correspondientes magnitudes del color y del

sonido anteriormente explicadas, asociando los colores oscuros a sonidos fuertes y los colores claros a sonidos suaves.

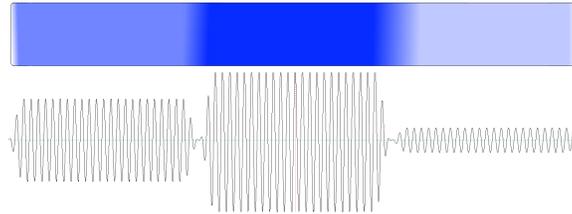


Figura 2: Relación entre la percepción continua del brillo del color y la intensidad del sonido

Adicionalmente, existe un sistema de traducción propuesto por Lagresille [40] basado en el fenómeno de la sinestesia, que establece una relación entre las propiedades del color y las del sonido. Se puede observar un esquema gráfico de esta traducción en la Fig. 3. La relación que se establece entre las propiedades del color y el sonido es la siguiente:

- **Matiz - Nota:** Para realizar esta traducción, el círculo cromático se divide en 12 secciones, haciendo coincidir cada una de ellas con la nota de la escala cromática correspondiente en el sistema de Lagresille [40].
- **Saturación - Volumen:** La saturación del color y el volumen de su nota correspondiente son directamente proporcionales. A mayor saturación del color, más intenso será el sonido correspondiente, es decir, más alto será su volumen, y viceversa.
- **Brillo - Octava:** Los valores de brillo del color y lo aguda que será la nota correspondiente son directamente proporcionales, es decir, cuanto más brillante sea un color determinado, más aguda será su nota correspondiente.

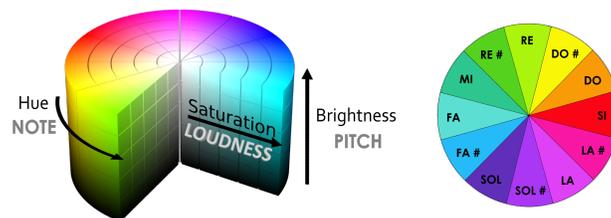


Figura 3: Relación entre el matiz del color y la nota de la escala cromática musical

Existe, además, otra línea de investigación en la cual la sinestesia se ha utilizado como una herramienta complementaria para la creación musical, y esta es la sonificación [1]. Se trata de un fenómeno en el cual se representa la información utilizando el sonido para lograr una mayor comprensión de un conjunto de datos. En [14] se utiliza la sonificación para realizar una representación auditiva del conjunto de datos, y en [33] se presenta una aplicación para que las personas invidentes puedan tener una percepción sensorial de una imagen.

El sistema de traducción color-sonido que se utilizará en este trabajo es el propuesto por Lagresille por la relación sencilla que establece entre las propiedades fundamentales de los mismos. Además, esta teoría se sustenta sobre la base del fenómeno de la sinestesia, el cual tiene un recorrido histórico muy amplio en el mundo de la creación artística. Adicionalmente se incluirán aspectos de la teoría de la sonificación para una mejor comprensión del proceso de composición por parte del usuario.

#### 2.4. Técnicas de Composición Musical

Una de las ramas de la Inteligencia Artificial que está empezando a tomar peso actualmente es la relacionada con la creatividad computacional [38]. En el caso de la composición musical, son tres grandes grupos de técnicas los que se destacan en la literatura:

- **Técnicas estadísticas:** Este tipo de técnicas tiene su base en la teoría de la Estadística [12]. Por una parte destacan las gramáticas, que aprovechan la existencia de normas básicas en la armonía y establecen las reglas necesarias para la composición. Por otra parte, los trabajos que utilizan Modelos de Markov estudian una serie de piezas para establecer la probabilidad de transición entre las diferentes configuraciones musicales (notas o acordes), y en base a ello, posteriormente generan música aplicando las probabilidades calculadas. [32] propone un sistema de composición musical basado en gramáticas para el cual se tienen en cuenta diversas características musicales (tono, duración...). *The Continuator* [39] es un sistema interactivo que compone música intentando imitar un estilo determinado en tiempo real.
- **Técnicas bioinspiradas** Este tipo de algoritmos se basan en la simulación del comportamiento de sistemas naturales o sociales para conseguir objetivos no determinísticos [45]. Dentro de este tipo de técnicas destacan las siguientes:
  - **Inteligencia de Enjambre (*Swarm Intelligence*):** Basan su funcionalidad en las capacidades de algunos seres vivos sociales para resolver tareas en su vida cotidiana [25]. Existen varios algoritmos de este tipo, como la Optimización de Colonia de Hormigas, que se basa en el rastro de feromonas que estos insectos dejan al caminar. [16] genera melodías armonizadas con estilo barroco. El algoritmo de Optimización del Enjambre de Partículas o *Particle Swarm Optimization* (PSO) trata de optimizar la distribución de una serie de partículas en un espacio. [42] obtiene motivos musicales basados en los movimientos de las partículas

en una ejecución de este algoritmo. Existen otros algoritmos de este tipo como el Algoritmo de la Luciérnaga o el Algoritmo del Murciélago.

- **Sistemas Inmunológicos Artificiales (AIS):** Permiten resolver problemas de optimización basándose en el funcionamiento del sistema inmune de los organismos vivos [13]. En [8] se realiza un proceso de síntesis de sonido mediante el uso de una red inmune artificial.
- **Algoritmos genéticos:** Se inspiran en la evolución biológica y en su base genético-molecular. Con ayuda de esta teoría, los algoritmos simulan la evolución de una población de individuos mediante su sometimiento a dos posibles acciones aleatorias similares a las que actúan en la evolución biológica (mutación y recombinación genética) y en la selección natural, al más puro estilo darwinista [17]. En [4] se propone un modelo en el cual se simula el proceso de aprendizaje de un músico improvisando jazz.
- **Técnicas basadas en agentes:** Un agente inteligente es una entidad capaz de percibir su entorno, y procesar la información de manera que responde o actúa de manera racional, buscando siempre cumplir un objetivo. Cuando existen varios agentes, cada uno de ellos realiza tareas de forma autónoma para lograr un objetivo común [15]. Existen propuestas en las que el usuario puede interactuar con el sistema multiagente, haciendo que sus decisiones influyan en el resultado final de la composición [27].

Tras este análisis se determina que la mejor opción para esta propuesta es la Inteligencia de Enjambre, y, más concretamente, el algoritmo PSO. Se trata de un algoritmo que se adapta al propósito del problema, ya que se trata de evaluar una serie de partículas cercanas a una dada y generar una función de optimización teniendo en cuenta la atracción con todas ellas. En este trabajo, las partículas representarán notas, y tendrán una posición aleatoria en el espacio y una velocidad constante que variará en base a la atracción o repulsión que sufra por parte de sus partículas más cercanas.

## 2.5. Visualización y Analítica Visual

El principal objetivo de la visualización y el análisis de datos es permitir que estos sean más comprensibles para el usuario, reduciendo la carga cognitiva y la dificultad de comprensión intrínseca al tema que se está tratando.

*Neuronal Synchrony* [7] realiza una aproximación entre la música y su concepción por una persona sinestésica, representando, en tiempo real, una animación que muestra la percepción de la música en su cerebro. Otro ejemplo de este tipo de herramientas podría ser la conocida *Music Animation Machine* [28], que es fruto de un proyecto de Stephen Malinowski que busca un entendimiento de la grafía musical por parte del usuario mediante una representación visual. Se realiza una representación de las notas considerando la duración y frecuencia de las mismas y plasmando esto con líneas de diferentes longitudes y alturas respectivamente. En este caso no existe una traducción sinestésica del sonido al color como tal.

Debido al interés común de la sonificación y la visualización de facilitar la interpretación de los datos a los usuarios, existen varios estudios que relacionan

ambas teorías. Uno de ellos podría ser *SonART* [2], que permite realizar una creación musical a partir de un dibujo que se realiza en tiempo real, dando lugar a una sonificación y visualización interactiva. Otro ejemplo podría ser [22], que aplica la teoría de la sonificación y la analítica visual al movimiento del cuerpo humano. Por su adaptación y correspondencia con el objetivo de visualización de este trabajo, se considera que la aplicación de técnicas de sonificación junto con la posibilidad de interactuar con el sistema es la opción más adecuada para esta propuesta.

### 3. Antecedentes

Como ya se ha comentado anteriormente, parte de los objetivos de esta propuesta ya fueron abordados en un trabajo anterior [30]. En él nos habíamos planteado la composición de una melodía a partir de los colores predominantes de una imagen determinada. La Fig. 4 ofrece una descripción global de la propuesta.

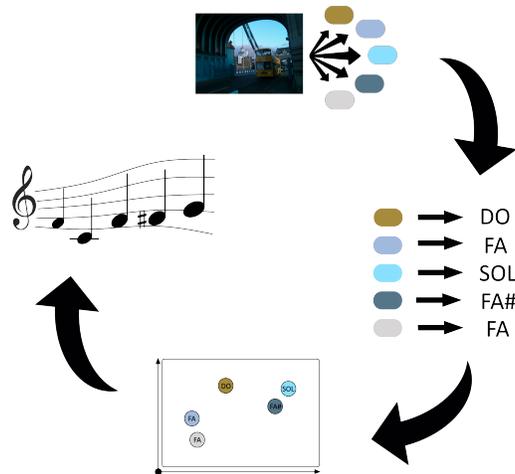


Figura 4: Visión global de la propuesta planteada en [30]

El primer paso era la detección y extracción de colores predominantes de la imagen, para lo que se utilizaba el algoritmo de cuantificación *Median Cut*. Estos colores se expresaban con el modelo de color HSV para identificar y cuantificar fácilmente sus principales propiedades, y a continuación se aplicaba el sistema de Lagresille [40] para traducir estas propiedades a cualidades típicas del sonido. Para poder representar estos sonidos numéricamente se hizo uso del formato estandarizado MIDI [41], y para el tratamiento de las propiedades musicales y armónicas del sistema tonal desde una perspectiva matemática se hizo

uso del TIS [3] y la DFT [6]. A continuación se aplicaba un algoritmo de optimización denominado *Particle Swarm Optimization* [25], que consideraba cada nota como una partícula situada inicialmente de manera aleatoria en un plano bidimensional. Estas partículas iban variando su posición en dicho espacio en base a la compatibilidad sonora que tuvieran con el resto de partículas de su entorno, calculada con la consonancia y la similitud entre notas. Finalmente, se establecía la tonalidad de la composición buscando la nota más repetida y decidiendo aleatoriamente el modo de la misma (mayor o menor), y se iniciaba un proceso de ordenación de notas por proximidad en el espacio y teniendo en cuenta otras propiedades musicales como la atracción melódica, la consonancia y la pertenencia a la tonalidad, que también eran calculadas con ayuda del TIS.

El producto de este trabajo fue la obtención de melodías que cumplieran con los criterios de la armonía occidental clásica de forma automática y con un alto grado de creatividad y flexibilidad en el proceso de composición. Además, los colores servían como base o fuente de inspiración para la composición, aunque es importante señalar el componente subjetivo que añadía la sinestesia en el proceso de traducción color-sonido. Sin embargo, la calidad de las melodías compuestas no era ideal, ya que los colores extraídos de la imagen determinaban e incluso condicionaban las notas con las que se realizaba la composición, dificultando la aplicación de las reglas de la teoría de la armonía clásica en el proceso creativo. Además, el algoritmo PSO suponía un coste computacional bastante elevado y el tiempo de espera para la obtención de la melodía era considerablemente alto.

## 4. Solución Propuesta

En esta sección se especifican los detalles de cada una de las etapas del sistema propuesto, comenzando por la extracción del color, pasando por la traducción de este al sonido (con su correspondiente notación y matemática) y el proceso de composición armónico, dejando para el final el modelo de visualización propuesto para que el usuario pueda comprender mejor todo el proceso.

### 4.1. Extracción del Color

Para el proceso de extracción del color, en esta propuesta se utiliza un método de cuantificación con la intención de reducir la cantidad de información a tratar, y con ello, la complejidad del problema. Concretamente, el algoritmo utilizado en este caso es el *Median Cut* debido a su robustez [20]. La aplicación de dicho algoritmo permite la obtención de una paleta con los 70 colores más representativos de la imagen, tal y como se puede observar en la Fig. 5. Se ha comprobado empíricamente que es un número suficiente para asegurar un mínimo de calidad musical en la composición sin llegar a ser excesivo para la aplicación de los algoritmos de optimización.

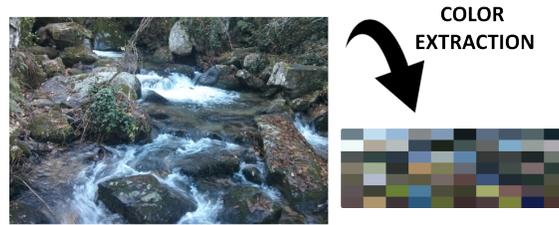


Figura 5: Proceso de extracción de los colores predominantes de la imagen

## 4.2. Traducción del Color en Sonido

Debido a su relación con el sistema visual humano de percepción de los colores, su facilidad de interpretación y la representación numérica de las propiedades del color, los colores extraídos se expresan en el formato HSV. Este modelo de color será además útil y necesario para la traducción del color a sonido, como se verá a continuación.

El proceso de traducción color-sonido hace uso del sistema de Lagresille. La aplicación de este sistema se realiza igual que en la propuesta anterior, de manera que el matiz del color da lugar a la nota, la saturación al volumen y el brillo a la octava.

Además, para lidiar con el problema de la subjetividad intrínseco a la sinestesia se realiza un estudio cuyos resultados se describen en la Sec. 5 para validar la elección en base a la aceptación de los usuarios.

## 4.3. Representación Matemática del Sonido

El formato MIDI es el elegido para recoger los sonidos en el paso inicial de esta propuesta. En la etapa final, por su sencillez de creación e interpretación de cara a un posible análisis formal de los resultados obtenidos, la generación de un archivo audible se realizará también con este formato.

Para los cálculos matemáticos de las propiedades musicales se hace uso de la teoría de [3], utilizando los vectores de croma definidos en la Sec. 2 para representar las configuraciones musicales deseadas, que en este caso no son notas sino acordes tríada. A estos vectores se les aplica la DFT, de donde se obtienen las frecuencias en forma de *Tonal Interval Vectors* (TIV) que serán utilizadas para calcular, matemáticamente, las características musicales deseadas que se especificarán más adelante (consonancia, similitud entre notas...). El espacio geométrico que se utilizará en este caso será el TIS, el cual permite, mediante el cálculo de distancias lineales y angulares, obtener representaciones numéricas de las propiedades musicales deseadas, que se especificarán a continuación.

#### 4.4. Proceso de Composición

En esta etapa se realiza la creación musical propiamente dicha, y, aunque también se hace uso del algoritmo PSO, supone un nuevo planteamiento respecto del trabajo previo.

En este trabajo, cada nota obtenida en el proceso de traducción color-sonido representa una región tonal que da lugar a un acorde triada, representado por una partícula en un espacio bidimensional tal y como ilustra la Fig. 6.



Figura 6: Relación entre el color, la nota, el acorde y la partícula

Una vez hallados todos los sonidos aislados se determina la tonalidad de la composición a partir de una teoría basada en las frecuencias (armónicos) [26].

Las partículas están dotadas de una velocidad inicial, y en cada iteración del algoritmo PSO, se evalúan de la siguiente manera:

- **Búsqueda de vecinos:** Mediante la distancia euclídea se buscan las partículas que se encuentran a un radio menor o igual que un cierto valor  $r$ .

$$distanciaEuclídea(Partícula_1, Partícula_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

- **Evaluación de vecinos:** Todos los acordes de cada tonalidad se agrupan en tres regiones tonales; a saber: Tónica (*I*, *III* y *IV* grados), Subdominante (*II* y *IV* grados) y Dominante (*V* y *VII* grados). La nota correspondiente a la partícula que se evalúa representa la fundamental de un acorde de triada, y cada vecino analizado se considera como la fundamental de otro; de estos acordes se obtienen las correspondientes regiones tonales. Posteriormente, se elige, aleatoriamente, un acorde de la región tonal en cuestión para lograr un mayor dinamismo, y se calcula el *fitness* entre ambos acordes. Para el cálculo de la función de *fitness* se tienen en cuenta cuatro conceptos fundamentales de la teoría de la armonía clásica:

- **Disonancia:** Evalúa lo desagradable que es la sensación de escuchar un grupo de sonidos de manera simultánea.
- **Similitud entre acordes:** Garantiza que dos acordes que vayan a sonar de manera secuencial sean mínimamente parecidos para lograr una evolución musical gradual.
- **Pertenencia a la tonalidad:** Considera la relación de un acorde concreto con la tonalidad establecida para la composición.

- **Distancia a la región tonal:** Garantiza que la variación entre regiones tonales no sea excesiva para que la sensación auditiva sea más estable.

Como muestra gráficamente la Fig. 7, estas características musicales se calculan matemáticamente de acuerdo con las ecuaciones siguientes:

$$Disonancia = distanciaEuclídea(Frecuencia_{acordes}, centro) \quad (2)$$

$$Similitud = distanciaEuclídea(Frecuencia_{acorde_1}, Frecuencia_{acorde_2}) \quad (3)$$

$$Tonalidad = ángulo(Frecuencia_{acorde_2}, Frecuencia_{tonalidad}) \quad (4)$$

$$Región = ángulo(Frecuencia_{acorde_{región}}, Frecuencia_{acorde_2}) \quad (5)$$

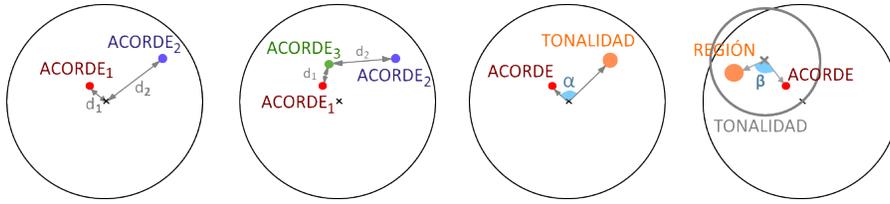


Figura 7: Definición gráfica de la disonancia, similitud entre acordes, pertenencia a la tonalidad y distancia a la región tonal respectivamente en el TIS.

Teniendo todos estos factores musicales en cuenta, el *fitness* se calcula como indica 6:

$$Fitness = W_d * Disonancia + W_s * Similitud + W_r * Región + W_t * Tonalidad \quad (6)$$

donde  $W_d$  es el peso de la disonancia,  $W_s$  el peso de la similitud entre acordes,  $W_r$  el peso de la región tonal y  $W_t$  el peso de la pertenencia del segundo acorde a la tonalidad. Los valores de estos parámetros se han establecido, empíricamente, como 3, 1, 1,5 y 5 respectivamente.

- **Actualización de la velocidad:** La nueva velocidad de la partícula viene determinada por su inercia, por su mejor posición en el espacio y por la atracción con el resto de partículas que están a una distancia relativamente pequeña:

- **Término de velocidad:** Representa la inercia, que es ponderada con un término amortiguador de la velocidad.

$$TérminoVelocidad = amortiguador * velocidadAnterior \quad (7)$$

- **Término propio:** Tiene en cuenta la diferencia entre la mejor posición de la propia partícula y su posición anterior, un coeficiente de aceleración cognitiva ( $c1$ ) y un valor aleatorio entre 0 y 1 ( $r1$ ).

$$TérminoPropio = c1 * r1 * (mejorPosición - posiciónAnterior) \quad (8)$$

- **Término social:** En este término se tiene en cuenta la atracción o repulsión que siente una partícula por las que están en su entorno cercano. Se pondera la distancia a cada una de ellas con la evaluación obtenida con la función de *fitness*. Además se utiliza un coeficiente de aceleración social ( $c2$ ) y un valor aleatorio entre 0 y 1 ( $r2$ ). Se establece una repulsión entre partículas para evitar colisiones cuando estas están muy próximas que se ha establecido empíricamente como 0,3.

$$Suma = \sum_{i=0}^{numVecinos} evaluación(i) * (posVecino(i) - posAnterior),$$

$$TérminoSocial = c2 * r2 * Suma,$$

$$TérminoSocial = TérminoSocial + (-TérminoSocial * repulsión) \quad (9)$$

- **Actualización de la posición:** Debido al cambio de velocidad, cada partícula se mueve en el espacio modificando su posición. Cada componente  $i$  de la nueva posición se halla por la suma de la posición anterior y la nueva velocidad, tal y como muestra 10.

$$PosiciónNueva_i = PosiciónAnterior_i + VelocidadNueva_i \quad (10)$$

La creación de la composición se va realizando mientras se ejecuta el algoritmo *swarm*. El primer acorde viene definido por una partícula que representa a una nota coincidente con la tónica de la tonalidad, elegida aleatoriamente. Cuando la última partícula de la composición permanece suficientemente junta a otra como para ser vecinas en  $n$  iteraciones consecutivas, se considera que su compatibilidad sonora es suficientemente buena y se añade la segunda a la composición. Este número de iteraciones consecutivas  $n$  se establece empíricamente, siendo 7 el valor ideal. El proceso se realiza de manera iterativa, hasta conseguir una progresión con un mínimo de  $m$  acordes o hasta que se ejecutan  $p$  iteraciones del algoritmo PSO cuyos valores se han definido, de manera empírica, como 10 y 250 respectivamente. Las composiciones siempre acaban en un acorde de la región tonal de Dominante seguido de un acorde de tónica. Una vez que la progresión alcanza la longitud máxima establecida, esta puede acabar por dos motivos:

- Se añaden a la progresión dos acordes que cumplen con los criterios de cadencia final especificados anteriormente (los correspondientes a los grados  $V - I$  o  $VII - I$  de la tonalidad).
- Se alcanza el número máximo de iteraciones establecido, en cuyo caso la cadencia de acordes  $V - I$  se añade automáticamente a la composición.

#### 4.5. Visualización

El objetivo de la aplicación de teorías de visualización en esta propuesta es la reducción de la carga cognitiva del usuario de manera que, tanto los datos como su evolución durante el proceso de composición, sean más comprensibles. Adicionalmente, estas técnicas se combinarán con algunas otras relativas a la sonificación.

Para la visualización de esta propuesta se ha decidido adaptar la herramienta de Gapminder (*Wealth and Health of nations*) [5]. A pesar de que la temática de esta herramienta y la de nuestro trabajo son distintas y no comparten, en principio, puntos de interés, son muchas las similitudes conceptuales que se pueden aprovechar. En Gapminder, cuyo objetivo es el estudio de la evolución de la economía y la salud en cada país y a nivel global, se representan todas las naciones mediante círculos de distintos tamaños y colores, de forma que el color indica el continente y el radio representa, de forma proporcional, el tamaño de su población.

En la adaptación para este trabajo, los puntos representan las diferentes partículas que se sitúan en un plano bidimensional con una distribución espacial inicial aleatoria variable debido al movimiento de las mismas. El color de cada partícula sería el que dio lugar a la nota antes de realizar la traducción color-sonido, dejando evidencia de la traducción sinestésica de Lagresille. Por otra parte, el tamaño de los puntos sería útil para representar la frecuencia de cada nota, de manera que las octavas más graves tendrían asociados puntos más grandes, y las octavas más agudas, puntos con un radio menor. Adicionalmente, el componente de la herramienta que se podría utilizar como eje entre el uso que proponía Gapminder y el de esta propuesta, es la dinamicidad existente en ambos casos. En la herramienta original, los datos evolucionaban con el paso del tiempo, y en esta propuesta, los datos evolucionan con cada iteración del algoritmo propuesto para la composición (PSO). El proceso de composición se podrá comprender de una manera visual (mediante una línea negra que une las partículas del fragmento) y auditiva (con la reproducción del mismo de acuerdo con la teoría de la sonificación). La Fig. 8 muestra la la distribución espacial de las partículas en la quinta iteración de una ejecución del algoritmo con 15 colores.



Figura 8: Visualización de dos iteraciones del PSO para un ejemplo de 15 partículas

Durante el tiempo ejecución del PSO el usuario está viendo qué está sucediendo y cómo se va generando la composición, aumentando su entendimiento sobre el proceso, reduciendo su carga cognitiva y amenizando el tiempo de espera. Además, una vez terminado el proceso de composición, la visualización de la composición queda a disposición del usuario para realizar un análisis de datos visual. Por este motivo, esta herramienta proporciona una serie de funcionalidades de interacción tales como el conocimiento de la nota que representa cada partícula al pasar el ratón por encima de ella y el sonido de la misma al presionarla con el ratón.

## 5. Resultados y Discusión

El objetivo principal de este trabajo era desarrollar un sistema creativo que elaborara una composición musical realizada de manera automática a partir de los colores de una imagen. En un trabajo previo ya habíamos abordado parcialmente la idea, desarrollando la primera fase del sistema, relativa al tratamiento de la imagen y la extracción de color, y aplicándola a la generación de una melodía simple. Desde ese punto de partida, este trabajo pone el foco en la parte más creativa.

La Fig. 9 muestra los resultados obtenidos en cada etapa del sistema para un ejemplo con 20 colores. En primer lugar, se realiza la extracción de los colores de la imagen, y a continuación tiene lugar la traducción de estos a sonidos. Posteriormente, cada nota es asociada a una partícula que es, a su vez, situada aleatoriamente en un espacio bidimensional delimitado. Después se determina la tonalidad de la composición [26], y, en base a ello, se calcula la región tonal de cada una de las partículas. A continuación comienza la ejecución del algoritmo PSO; en la Fig. 9 se muestra la distribución de las partículas en la iteración número 22, donde coincide que se añade un acorde a la composición remarcando y uniendo las partículas que dan lugar a los acordes y reproduciendo el fragmento, y en la 74. Finalmente, se obtiene la progresión de acordes.

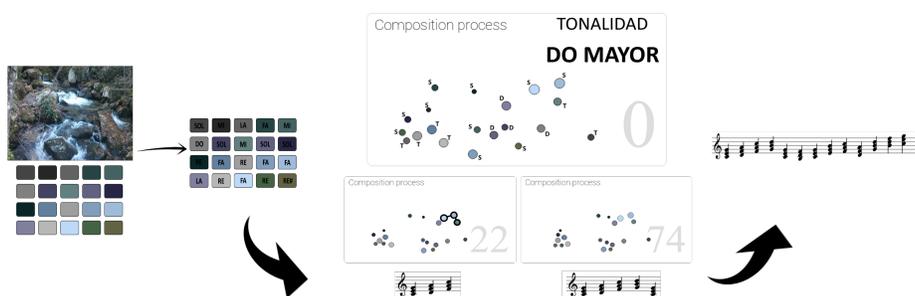


Figura 9: Visión global de las etapas del sistema propuesto

La evaluación del sistema se va a realizar desde dos puntos de vista diferentes: por una parte se va a evaluar la calidad de la traducción color-sonido aplicada, y por otra, se se va a analizar la composición musical obtenida.

### 5.1. Validación del Sistema de Traducción Color-Sonido

En este apartado se va a evaluar la calidad de traducción matiz-nota, para comprobar si el sistema es aceptado por los usuarios o el componente subjetivo de la sinestesia afecta notablemente de forma negativa, haciendo que el sistema no se pueda aplicar a este problema. De esta manera se medirá la correlación entre los colores de la composición de una imagen digital y los sonidos obtenidos con este sistema de traducción.

Para ello, se ha diseñado un test donde los usuarios deberán valorar la relación entre los colores de una imagen y los sonidos del fragmento que se ha compuesto para ella en base a su propia percepción. En este caso, los fragmentos utilizados son melodías; de esta manera será más sencillo evaluar la correlación en cuestión. En el enlace <https://form.jotformeu.com/70101890150342> se puede consultar el test. En este caso, 32 personas han realizado el test, con el objetivo de evaluar la similitud entre los colores y los sonidos en base a su propia percepción. Los resultados obtenidos para dicho test se pueden observar en la Fig. 10.

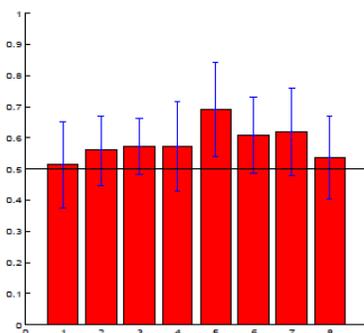


Figura10: Valoración de la relación entre el color y el sonido

El nivel de similitud entre la melodía y los colores extraídos ha sido valorada por encima de la media (por encima del 0,5). En el eje horizontal se representa cada ejemplo evaluado y, en el eje vertical, el rango de valores para la votación. Sin embargo, la desviación estándar muestra que algunas de las opiniones de los usuarios están por debajo del valor medio, considerando que la traducción color-sonido no es suficientemente buena. Sin embargo, se considera que, como el valor medio de las valoraciones para cada ejemplo está por encima de la media, y la percepción de similitud entre colores y sonido es subjetiva, el sistema de traducción de color-sonido propuesto por Lagresille es adecuado para este

problema. No se puede perder de vista el hecho de que no se trata de realizar una composición musical que describa la imagen, sino que esté inspirada en sus colores de alguna manera.

## 5.2. Análisis de la Etapa de Generación de Armonía

En este apartado se van a analizar, por una parte, las mejoras que introduce la adaptación del algoritmo PSO en este trabajo frente al algoritmo propuesto en [30], y por otra, aspectos musicales de los resultados obtenidos. Tras la obtención de los sonidos aislados se determina la tonalidad de la composición. Posteriormente se aplica un algoritmo de optimización (PSO) con el objetivo de obtener un agrupamiento de las notas en el espacio en base a su consonancia, la similitud entre acordes, la distancia a la región tonal y la pertenencia a la tonalidad establecida. Durante esta redistribución de partículas en el espacio se va generando una progresión de acordes. El resultado es una composición musical con una complejidad notable (se obtiene una creación armónica), cumpliendo las reglas de la armonía occidental clásica. Además se introducen técnicas de visualización y sonificación para que el usuario comprenda mejor el “porqué” y el “cómo” de la tarea creativa que se desempeña.

La Tab. 2 muestra una serie de datos para cada una de las propuestas que evidencian la consecución de unos resultados significativamente mejores en este trabajo.

	Propuesta anterior	Propuesta presente
Fases de composición	5	4
Iteraciones PSO	500	250
Tiempo de composición	9'	3,2'
Complejidad musical	Melodía	Armonía
Determinismo notas	Alto	Medio
Dinamismo notas	Nulo	Medio
Técnicas de analítica	-	Visualización y sonificación

Tabla 2: Características de la etapa de composición en este trabajo y en el anterior

En el caso de la propuesta anterior, tras ordenar las partículas en el espacio atendiendo a su compatibilidad sonora, se realizaba un proceso de ordenación de notas que daba lugar a la melodía. En este caso esto no es necesario. El número de iteraciones del PSO también se reduce considerablemente, disminuyendo así el coste computacional casi un 60%. Por otra parte, la composición obtenida en la primera propuesta era una melodía sencilla, y en este enfoque se genera una progresión de acordes. La traducción color-sonido se realiza de manera fija

en ambos casos; en la primera propuesta estas notas eran utilizadas sin sufrir variaciones, por lo que la composición musical venía determinada por los colores de la imagen. Sin embargo, en el caso de la segunda propuesta, al añadir el concepto de “región tonal” y la elección del acorde de manera aleatoria, el determinismo se ve menguado y existe una mayor variación de acordes disponibles para la composición. Por último, en la propuesta actual se incluyen técnicas de visualización y sonificación para facilitar la comprensión del proceso.

Para evaluar los resultados desde un punto de vista musical se diseña un test de usuario en el que se deben valorar una serie de parámetros que darán evidencia de la calidad del sistema. El test, que se puede encontrar en el siguiente enlace: <https://form.jotformeu.com/71934490518361>, ha sido respondido por 25 expertos en música tras probar el sistema. Estos usuarios han valorado los resultados obtenidos con una puntuación del 0 al 10. En la Tab. 3 se muestra el valor medio de las puntuaciones de los usuarios para cada uno de los aspectos evaluados:

	Valoración media
Complejidad musical	7,64
Calidad musical	7,84
Calidad armónica	8,52
Comprensión del proceso	6,84
Valoración experiencia	9,48

Tabla 3: Valoración de expertos musicales para analizar la calidad del sistema.

Para el caso de la complejidad musical el valor medio de las puntuaciones de los usuarios es de 7,64. Esto es así porque el sistema realiza una creación armónica, y las reglas musicales para la realización de una composición armónica no son sencillas. La evaluación de la calidad musical, valorada con un 7,84, es bastante positiva, dejando evidencia de que los factores armónicos elegidos para la función de optimización son adecuados y dan lugar a resultados numérica y musicalmente muy buenos. Por este mismo motivo se justifica que la evaluación de la calidad armónica sea de 8,52; las composiciones obtenidas por el sistema cumplen con las reglas de la armonía occidental clásica gracias a los factores musicales considerados en el proceso de composición. La comprensión del proceso recibe una puntuación de 6,84. A pesar de ser el factor con una puntuación más baja es un resultado muy positivo si se tiene en cuenta la dificultad del proceso. Por último, la valoración de la experiencia con el sistema ha sido óptima.

Con estos resultados se podría afirmar que se han conseguido los objetivos fijados. Por una parte, la utilización del sistema de Lagresille permite realizar una traducción color-sonido que tiene un alto grado de aceptación, de manera general, por parte de los usuarios. Adicionalmente, el enfoque para la composición musical es muy novedoso y los resultados obtenidos son, musicalmente, satisfactorios. Por

otra parte, la reducción de tiempo de computación y el uso de técnicas como son la visualización y la sonificación influyen en la consecución de una buena experiencia de usuario.

## 6. Conclusiones

En este trabajo, proponíamos un sistema que fuera capaz de realizar una composición musical armónica inspirada en una imagen, utilizando una traducción sinestésica de sus colores. Para ello se han aplicado reglas de la armonía tonal occidental.

En lo relativo al tratamiento de la imagen, la técnica de cuantificación del color posibilita la reducción de colores que forman una imagen digital sin pérdidas de información relevante para el trabajo. Por otra parte, el modelo del color HSV facilita la comprensión de las características principales del mismo y la aplicación del sistema de Lagresille para la traducción color-sonido.

Para la representación musical, el estándar MIDI permite la representación de los sonidos aislados y de la composición final. Sin embargo, las relaciones aritméticas que se producen entre los números asignados y las notas musicales no tienen ningún tipo de relación con las propiedades de la armonía necesarias para la composición. Por este motivo es necesaria la combinación de este estándar con otras técnicas más complejas como son el modelo geométrico *Tonal Interval Space* y el cálculo de la Transformada Discreta de Fourier.

El proceso de composición no es una espera tediosa para el usuario en este caso. El tiempo de ejecución es muy aceptable debido a la rápida convergencia del algoritmo evolutivo. Uno de los factores que más influyen en esto es el diseño de la función de *fitness* y la elección de las propiedades musicales que se consideran en ella. Además, las técnicas de visualización y sonificación utilizadas hacen que la espera sea mucho más entretenida y permiten que el usuario comprenda en cierto grado el proceso de composición a pesar de su complejidad.

Por último, los análisis de los resultados son muy positivos. La validación del sistema de traducción del color al sonido se realiza con un test, cuyos resultados demuestran que el nivel de similitud entre las notas musicales y los colores extraídos de la imagen es aceptable. Aunque no hay que perder de vista que el sistema no pretende describir una imagen sino buscar inspiración en ella para realizar el proceso de composición. Por otra parte, los resultados obtenidos con esta propuesta son sustancialmente mejores que los de la propuesta antecedente. El test realizado a expertos en música evidencia que la calidad musical de las composiciones es buena.

### 6.1. Líneas de Trabajo Futuro

Teniendo en cuenta los resultados del test de aceptación del modelo de traducción color-sonido se pueden explorar otras posibilidades para mejorar la propuesta de Lagresille y reducir el nivel de subjetividad de esta etapa. Algunas propuestas son la realización de un estudio social con personas sinestésicas para

intentar establecer una relación entre el matiz del color y la nota, y la aplicación de varios modelos de traducción de estas características de manera probabilística.

En lo relativo a la música, sería interesante estudiar cómo mejorar algunos aspectos. El primero de ellos es considerar las inversiones de los acordes (alterar el orden de las notas de los mismos). Además, también sería interesante trabajar con acordes de más de 3 notas para enriquecer aún más la composición. Por otra parte, sería deseable añadir la dimensión rítmica a las composiciones obtenidas.

Adicionalmente, el usuario podría elegir un determinado estilo musical para la imagen que proporciona. De esta manera, las notas de la composición vendrían dadas por la imagen, pero el estilo de la composición sería variable. Para ello se podría hacer uso de algunas técnicas como los Modelos de Markov.

## Referencias

1. S. Barrass and G. Kramer, "Using sonification," *Multimedia systems*, vol. 7, no. 1, pp. 23–31, 1999.
2. O. Ben-Tal, J. Berger, B. Cook, M. Daniels, and G. Scavone, "Sonart: The sonification application research toolbox." Georgia Institute of Technology, 2002.
3. G. Bernardes, D. Cocharro, M. Caetano, C. Guedes, and M. E. Davies, "A multi-level tonal interval space for modelling pitch relatedness and musical consonance," *Journal of New Music Research*, vol. 45, no. 4, pp. 281–294, 2016.
4. J. A. Biles, "Genjam: A genetic algorithm for generating jazz solos," in *ICMC*, vol. 94, 1994, pp. 131–137.
5. M. Bostock, "The wealth and health of nations," <https://bost.ocks.org/mike/nations/>, 2012, [Online; accessed 03-June-2017].
6. R. N. Bracewell and R. N. Bracewell, *The Fourier transform and its applications*. McGraw-Hill New York, 1986, vol. 31999.
7. J. Brandel, "Neuronal synchrony," <https://github.com/jonobr1/Neuronal-Synchrony>, 2013, [Online; accessed 28-May-2017].
8. M. Caetano, J. Manzolli, and F. Von Zuben, "Application of an artificial immune system in a compositional timbre design technique," *Artificial Immune Systems*, pp. 389–403, 2005.
9. E. Chew, "Out of the grid and into the spiral: Geometric interpretations of and comparisons with the spiral-array model." *Computing in musicology*, vol. 15, 2007.
10. R. Cohn, "Neo-riemannian operations, parsimonious trichords, and their tonnetz-representations," *Journal of Music Theory*, vol. 41, no. 1, pp. 1–66, 1997.
11. F. Collopy, "Color, form, and motion: Dimensions of a musical art of light," *Leonardo*, vol. 33, no. 5, pp. 355–360, 2000.
12. D. Conklin, "Music generation from statistical models," in *Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*. AISB Society London, 2003, pp. 30–35.
13. L. N. De Castro and J. Timmis, "An artificial immune network for multimodal function optimization," in *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on*, vol. 1. IEEE, 2002, pp. 699–704.
14. R. T. Dean, M. Whitelaw, H. Smith, and D. Worrall, "The mirage of real-time algorithmic synaesthesia: Some compositional mechanisms and research agendas in computer music and sonification," *Contemporary Music Review*, vol. 25, no. 4, pp. 311–326, 2006.

15. J. Ferber, *Multi-agent systems: an introduction to distributed artificial intelligence*. Addison-Wesley Reading, 1999, vol. 1.
16. M. Geis and M. Middendorf, "An ant colony optimizer for melody creation with baroque harmony," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. IEEE, 2007, pp. 461–468.
17. D. E. Goldberg, *Genetic algorithms*. Pearson Education India, 2006.
18. S. Harnad, "Categorical perception," 2003.
19. C. Harte, M. Sandler, and M. Gasser, "Detecting harmonic change in musical audio," in *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*. ACM, 2006, pp. 21–26.
20. P. Heckbert, *Color image quantization for frame buffer display*. ACM, 1982, vol. 16.
21. N. A. Ibraheem, M. M. Hasan, R. Z. Khan, and P. K. Mishra, "Understanding color models: a review," *ARPN Journal of Science and Technology*, vol. 2, no. 3, pp. 265–275, 2012.
22. A. R. Jensenius, "Motion-sound interaction using sonification based on motion-grams," 2012.
23. K. Jo and N. Nagano, "Monalisa:"see the sound, hear the image"." in *NIME*, 2008, pp. 315–318.
24. L. Kang, T. Gu, and G. Gay, "Harmonic paper: interactive music interface for drawing," in *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2013, pp. 763–768.
25. J. Kennedy, "Swarm intelligence," in *Handbook of nature-inspired and innovative computing*. Springer, 2006, pp. 187–219.
26. C. L. Krumhansl, "Tonal hierarchies and rare intervals in music cognition," *Music Perception: An Interdisciplinary Journal*, vol. 7, no. 3, pp. 309–324, 1990.
27. M. L. Maher, "Computational and collective creativity: who's being creative?" in *ICCC*, 2012, pp. 67–71.
28. S. Malinowski and L. Turetsky, "Music animation machine," 2011, [Online; accessed 30-May-2017].
29. L. E. Marks, "On associations of light and sound: The mediation of brightness, pitch, and loudness," *The American journal of psychology*, pp. 173–188, 1974.
30. L. Martín-Gómez, M. Navarro-Cáceres, and F. De la Prieta, "Plataforma para la creación de melodías," Trabajo Fin de Grado en Ingeniería Informática, 2016, universidad de Salamanca.
31. M. Á. Mateu, *Armonía práctica*. Ab música, 2004.
32. J. McCormack, "Grammar based music composition," *Complex systems*, vol. 96, pp. 321–336, 1996.
33. M. Mengucci, F. Medeiros, and M. Amaral, "Image sonification application to art and performance," in *proceedings of INTER-FACE International Conference on Live Interfaces, Lisbon*, 2014.
34. O. Messiaen, *Chronochromie: pour grand orchestre*. A. Leduc, 1963.
35. O. Messiaen, P. Boulez, and Y. Loriod, *Couleurs de la cité céleste*. Leduc, 1966.
36. A. H. Munsell *et al.*, "Munsell book of color," 1950.
37. M. Navarro, M. F. Caetano, G. Bernardes, L. N. de Castro, and J. M. Corchado, "Automatic generation of chord progressions with an artificial immune system." in *EvoMUSART*, 2015, pp. 175–186.
38. M. Navarro, J. M. Corchado, and M. Freitas, "Sociedades humano-agente: Un caso de estudio en creatividad musical," Tesis Doctoral - Departamento de Informática y Automática, 2017, universidad de Salamanca.

39. F. Pachet, "The continuator: Musical interaction with style," *Journal of New Music Research*, vol. 32, no. 3, pp. 333–341, 2003.
40. J. C. Sanz, *Lenguaje del color:(sinestesia cromática en poesía y arte visual)*. El autor, 1981.
41. E. Selfridge-Field, *Beyond MIDI: the handbook of musical codes*. MIT press, 1997.
42. J. Serrà and J. L. Arcos, "Particle swarm optimization for time series motif discovery," *Knowledge-Based Systems*, vol. 92, pp. 127–137, 2016.
43. I. Sysoev, R. D. Chitloor, A. Rajaram, R. S. Summerlin, N. Davis, and B. N. Walker, "Middie mercury: an ambient music generator for relaxation," in *Proceedings of the 8th Audio Mostly Conference*. ACM, 2013, p. 20.
44. J. W. Von Goethe, *Theory of colours*. Mit Press, 1840, vol. 3.
45. X.-S. Yang, *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.

# Sistemas Inteligentes para la Resolución de Problemas en el Campo de la Biotecnología

Yeray Mezquita Martín<sup>1</sup> y Emilio Santiago Corchado Rodríguez<sup>1</sup>

Departamento de Informática y Automática, Facultad de Ciencias  
Plaza de los Caídos s/n, 37008, Salamanca, España  
{yeraymm,escorchado}@usal.es

**Resumen** La necesidad de diagnosticar a tiempo enfermedades genéticas como el cáncer, Alzheimer o ELA, ha hecho que los investigadores vuelquen sus esfuerzos en el desarrollo de sistemas inteligentes capaces de distinguir si una persona las padece o no. Estos sistemas pueden trabajar con los datos de la expresión genética de pacientes obtenidos de *microarrays*. En este trabajo se desarrolla un sistema inteligente capaz de seleccionar un subconjunto de genes que permita realizar predicciones sobre esos datos. Con él se busca obtener un equilibrio entre el número de genes utilizados y la precisión de las predicciones. También se comparan los resultados obtenidos, para tres conjuntos de datos, con los obtenidos por el método de filtrado por ganancia de información para esos mismos conjuntos. Se demuestra que el sistema propuesto mejora la precisión utilizando menos genes para realizar la clasificación en la mayoría de experimentos.

**Keywords:** Machine learning, selección de genes, correlación, information gain, clustering, predicción, microarrays.

## 1. Introducción

Hoy día, uno de los principales problemas del primer mundo es encontrar la forma de hacer frente a la proliferación de nuevos casos de enfermedades genéticas complejas, tales como el cáncer [26] o el Alzheimer [1]. Cada uno de estos casos requiere de un diagnóstico temprano, necesario para comenzar el tratamiento durante sus primeras etapas, lo que ayuda a evitar complicaciones que encarezcan dicho tratamiento y pongan en riesgo la salud del paciente.

Encontrar la forma de distinguir pacientes que padezcan una de estas enfermedades y aquellos que no, de forma precisa y a tiempo, puede ayudar a salvar la vida de millones de personas en todo el mundo además de ahorrar una gran cantidad de dinero por las complicaciones de los tratamientos al realizarse sobre pacientes en fases avanzadas de las enfermedades.

Este tipo de enfermedades se caracterizan por aparecer debido a desórdenes en la síntesis de proteínas de las personas, ocurridos por el mal funcionamiento de algunos genes. Por ello, una de las formas que permiten realizar la distinción entre personas sanas y enfermas es mediante el uso de clasificadores que utilicen datos de cómo trabaja el genoma de estas personas.

Una de las formas que existen actualmente de medir el funcionamiento de los genes, es la utilización de chips de ADN, también llamados *microarrays*. Dentro de los tipos de conjuntos que pueden ser obtenidos de los *microarrays*, en este trabajo, se van a utilizar aquellos que dividen las muestras del conjunto en dos clases.

El artículo se divide de la siguiente manera: en la sección 2 se hace una pequeña revisión del estado del arte, en la sección 3 se presenta el sistema inteligente propuesto, en la sección 4 se exponen y comparan los resultados obtenidos por el sistema desarrollado y por el filtrado por ganancia de información y, por último, en la sección 5 las conclusiones y líneas de trabajo futuras.

## 2. Revisión del estado del arte

### 2.1. Datos de expresión genética

Una de las características de estos datos es que suelen ser ruidosos y contener valores perdidos (NAs, del inglés *Not A number*).

En [4] se evalúa el rendimiento de 9 algoritmos de estimación de valores perdidos sobre 13 conjuntos de datos, utilizando 3 tipos de medidas distintas en cada evaluación. En él se llega a la conclusión de que los métodos basados en *local least squares* son las mejores elecciones para la mayoría de los conjuntos de datos utilizados.

### 2.2. Algoritmos de selección de características

Tal como se muestra en los estudios realizados por [19], [3], [8] y [16], las estrategias propuestas en la selección de las características más relevantes para la predicción de la clase de las muestras, pueden ser catalogadas como *filters*, *wrappers*, *embededs* o *ensembles*.

**Filters** Estos métodos realizan un ranking de las características en base a la puntuación obtenida por el algoritmo utilizado, sin tener en cuenta las propiedades intrínsecas de los datos. Tras ello, seleccionan un número determinado de las características mejor posicionadas o aquellas características que superen un valor umbral dado. Como ejemplos más utilizados están la prueba U de Mann-Whitney [17], prueba F del análisis de la varianza (ANOVA, de su nombre en inglés *ANalysis Of VAriance*) [7], ganancia de información [10].

**Wrappers** Estas técnicas se encargan de seleccionar el subconjunto de características más discriminante minimizando el error de predicción de un clasificador determinado.

**Embededs** Los métodos embebidos tratan de reducir el tiempo de computación necesario por los *wrappers*, evitando reclasificar diferentes subconjuntos de características. Para ello, realizan la selección de características dentro del proceso de entrenamiento del clasificador.

**Ensembles** Las técnicas de ensamblado tienen por objetivo mejorar la tarea de clasificación utilizando un conjunto de características más robusto. Para ello, primero se obtienen diferentes conjuntos de características al ejecutar en paralelo cada una de esas técnicas de selección. Más tarde, todos los conjuntos obtenidos, son fusionados en uno solo [2][20][13].

En [11], se muestra como también estas técnicas se pueden utilizar para la tarea de clasificación. Para ello se proponen varios clasificadores, generalmente un número impar, que hacen la tarea de predicción en paralelo. Con el fin de decidir a qué clase pertenece una muestra, se hace un conteo de los resultados obtenidos por los clasificadores, dando como resultado final, la clase que más aparezca en ellos. Este tipo de métodos mejora los resultados de los clasificadores utilizándolos por separado [25].

### 2.3. Algoritmos de clasificación

Estos algoritmos se encuentran dentro del marco de aprendizaje supervisado, los cuales consisten en un entrenamiento con un conjunto de instancias de datos correctamente clasificados. Generalmente se utiliza una parte del conjunto de datos para realizar dicho entrenamiento. El resto de los datos es utilizado para probar el clasificador entrenado, lo que permite realizar una validación de la precisión del mismo.

Según el estudio realizado en [18], los algoritmos de predicción se podrían dividir en las técnicas que utilizan agentes, denominadas inteligencia de enjambre, y las que no.

## 3. Sistema inteligente propuesto

El sistema inteligente propuesto tratará de filtrar las características menos representativas de los conjuntos de datos binarios con los que trabajará. Esto lo conseguirá buscando un equilibrio, entre minimizar el error de la predicción de la clase de cada muestra y la cantidad de sondas a utilizar en esa predicción. En la Fig.1 se puede observar un diagrama con los diferentes bloques y etapas que consta el sistema inteligente.

Una de las características del sistema propuesto, es que los algoritmos presentados en cada etapa pueden ser sustituidos por otros que aparezcan en el estado del arte, en caso de que futuros experimentos lo requieran porque aporten una mejora al proceso.

### 3.1. Bloque1: Preprocesado de los datos

Debido a que las variaciones técnicas de los chips utilizados para obtener un conjunto de datos, son capaces de ocultar la información real que aportan éstos, se hace necesario realizar un normalizado que permita compararlos. Para ello, en este trabajo se han utilizado las técnicas de:

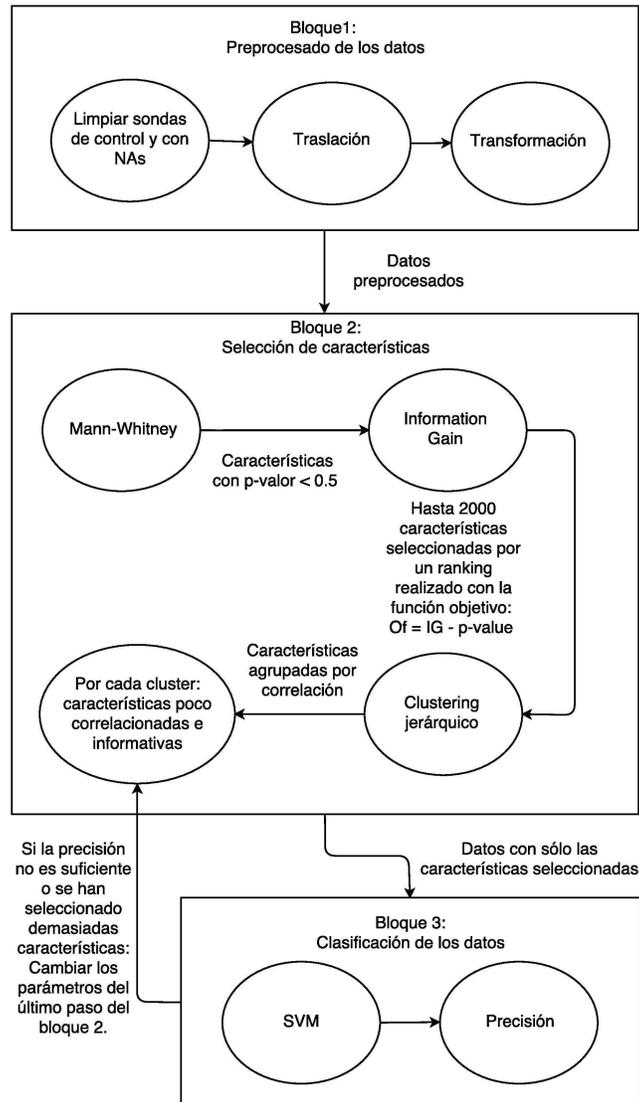


Figura 1: Diagrama del sistema inteligente propuesto.

- **Traslación:** con esta técnica se obtiene un centrado de los datos en el eje de coordenadas. Para ello, se resta la media aritmética de cada sonda, ecuación 1, a cada uno de sus valores.

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} \quad (1)$$

- **Transformación:** esta técnica permite evitar que los valores de una sonda enmascaren la información aportada por los de otra. Primero se obtiene la desviación estándar de cada sonda, ecuación 2, para posteriormente dividir cada uno de sus valores por su desviación estándar.

$$\sigma_j = \sqrt{\frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}{n}} \quad (2)$$

Además, para evitar que haya fallos durante la ejecución de estos algoritmos, se eliminan del conjunto todos aquellos genes que contengan datos con NAs.

### 3.2. Bloque2: Selección de características

**Filtrado 1: Prueba U de Mann-Whitney** Este filtrado es el encargado de eliminar todas aquellas sondas que aportan ruido o no son significativas a la hora de distinguir entre las dos condiciones en las que se dividen las muestras del dataset. Dado que se va a trabajar con conjuntos de datos en los que la distribución de los mismos no se conoce, se utilizará un filtrado no paramétrico, la prueba U de Mann-Whitney [17].

Si las medianas son muy diferentes  $U$  será próximo a 0, lo que indicará una diferencia estadística entre los valores de dichas poblaciones. Por el contrario, aquellas sondas que no tengan un valor umbral de  $U$  menor a 0,05 son eliminadas, ya que no ofrecen una diferencia estadística entre ambas poblaciones suficientemente grande como para considerarlas relevantes de estudio.

**Filtrado 2: Ganancia de información** Para este segundo paso del bloque de selección de características, se ha decidido utilizar el método de Ganancia de Información, algoritmo ampliamente utilizado en la literatura con el que realizar un ranking de las características. Además se ha decidido tener también en cuenta el valor obtenido por el paso anterior, para ello se busca maximizar una función objetivo utilizando tanto el valor del paso anterior como el de la ganancia de información.

Gracias al algoritmo de ganancia de información, se puede obtener una medida cuantitativa de la información aportada por cada atributo a la hora de realizar predicciones sobre la variable objetivo. Este cálculo se obtiene a partir de la entropía de Shannon [22]. Shannon la describió como la medida de impredecibilidad del estado, y viene definida por la fórmula de la ecuación 3:

$$H(N) = - \sum_{i=1}^k P(C_i, N) \log_2 P(C_i, N) \quad (3)$$

donde  $N$  es el número de muestras o instancias del conjunto total,  $k$  es el número de clases que puede dar como resultado la predicción,  $C_i$  es el conjunto de instancias de la clase  $i$ ,  $P(C_i, N)$  es la proporción de instancias que existen de la clase  $i$  respecto al número total de instancias, véase la ecuación 4.

$$P(C_i, N) = \frac{C_i}{N} \quad (4)$$

Para obtener la entropía que posee una determinada característica en el conjunto de datos se utiliza la fórmula descrita por la ecuación 5:

$$H_\gamma(N) = \sum_{j=1}^m \frac{N_j}{N} H(N_j) \quad (5)$$

donde  $\gamma$  es la característica a evaluar, con  $v = \{v_1, v_2, \dots, v_m\}$  valores distintos, y  $N_j$  es el número de muestras que poseen el valor  $v_j$  de la característica  $\gamma$ .

La ganancia de información que aporta la característica  $\gamma$  es la diferencia entre la entropía total del conjunto y la obtenida por la característica, tal como se muestra en la ecuación 6. De esto se deduce que entre menor sea la entropía de la característica, mayor será la información que aporta, lo que repercute en un mayor poder de predicción por dicha característica.

$$IG(\gamma) = H(N) - H_\gamma(N) \quad (6)$$

La meta de este filtrado es realizar un ranking de características más fiable al utilizar más de un criterio. Para ello se combinan los resultados obtenidos en el paso anterior con los obtenidos con este algoritmo, maximizando una función objetivo con los p-valores de la fase anterior, valores pequeños, y las ganancias de información obtenidas en esta, valores grandes. Esta función se define en la ecuación 7:

$$OF(\gamma) = IG(\gamma) - pvalor_\gamma \quad (7)$$

siendo  $pvalor_\gamma$  el p-valor obtenido en la fase anterior para la sonda  $\gamma$ .

Con este ranking, en caso de ser necesario, se realiza un filtrado que deje hasta un máximo de 2000 sondas en el conjunto de datos. Lo que permite reducir los costes de los algoritmos que se ejecutarán en los siguientes pasos del sistema inteligente propuesto sin renunciar a trabajar con información que puede ser relevante.

**Clustering jerárquico** Tras obtener un número razonable de características que permitan trabajar con un menor coste sin asumir pérdida de información, se prosigue realizando grupos o *clusters* de dichas características. El fin de este paso es obtener en cada grupo aquellas características que aportan información redundante. Las características seleccionadas para cada grupo son las más cercanas entre sí según una medida de distancia seleccionada. En este trabajo se ha escogido la distancia euclídea, métrica utilizada para medir la línea recta que

separa dos puntos, en este caso las características del conjunto de datos, aunque puede ser sustituida por otra medida de distancias. La distancia euclídea entre dos características se mide como si ambas fueran puntos en el espacio de coordenadas, siendo cada una de las muestras coordenadas en el espacio.

Para realizar la tarea asignada en este paso, y dado que no se ha analizado de antemano la estructura de los datos a utilizar en este sistema inteligente, se ha optado por el uso de un algoritmo de clustering jerárquico. Este es un método de clustering que se encarga de construir una jerarquía de *clusters*, calculando la distancia que existe entre las características en base a sus valores en el conjuntos de datos.

Una vez se conoce el árbol de jerarquías, se prosigue con la obtención de los *clusters* que son óptimos para el conjunto de datos. Estos se pueden obtener mediante poda con un valor umbral para las distancias. Sin embargo, debido a las características de este trabajo, se ha optado por utilizar un algoritmo que realiza esta separación de *clusters* de forma automática [14].

**Filtrado 3: Selección por correlación** Gracias al paso anterior, en cada *cluster* están agrupadas las características más redundantes entre sí. Por esto, la selección que se va a realizar en este paso se realizará por cada grupo. Ésta consiste en obtener todas aquellas características suficientemente informativas por sí mismas y que no sean redundantes entre sí, realizando los siguientes pasos por cada *cluster*:

- Utilizando el método de correlación de Pearson, se obtiene una matriz de correlaciones entre todas las características presentes en el *cluster*. Este método viene definido por la covarianza de dos variables,  $X$  e  $Y$ , como se muestra en la ecuación 8, dividida por el producto de sus desviaciones estándar, véase la ecuación 2, de la forma en que se observa en la ecuación 9:

$$\text{cov}(X, Y) = E[(X - E[X])(Y - E[Y])] \quad (8)$$

donde  $E[X]$  es la media aritmética de la variable  $X$ , véase la ecuación 1.

$$\rho_{X,Y} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} \quad (9)$$

- De la matriz se obtienen los valores absolutos, con el fin de tener en cuenta sólo la correlación existente entre los elementos, y no si es directa o inversa.
- Para un umbral de correlación definido, se van seleccionando pares de características que no rebasen dicho umbral.
- Puesto que se pretende tener en cuenta también todos los pasos anteriores realizados por el sistema inteligente, se utiliza un umbral de información para los valores de la función objetivo. Con esto se consigue que sólo aquellas características que aportan una información mínima sean seleccionadas, descartando las que, aunque no sean redundantes entre sí, no aportan suficiente información útil para realizar la clasificación.

Gracias al clustering realizado en el paso anterior, este proceso resulta menos costoso. Esto se debe a que en cada paso del proceso se realiza sobre el subconjunto de sondas perteneciente a un cluster y no el todo conjunto completo de datos.

### 3.3. Bloque3: Clasificación de los datos

Para validar el conjunto de características seleccionadas hasta el momento, en este trabajo de fin de máster se ha decidido utilizar una máquina de soporte vectorial. Las máquinas de soporte vectorial, *Support Vector Machines* o *Support Vector Networks* [5], son algoritmos de clasificación de aprendizaje supervisado formalmente definidos por un hiperplano de separación, el cual es capaz de resolver los problemas de clasificación entre dos clases distintas. Este hiperplano, denominado óptimo, es aquel que maximiza los márgenes que existen entre él y el conjunto de entrenamiento.

En caso de que la clasificación realizada no sea considerada suficientemente precisa, se cambian los parámetros de corte del último paso del bloque de selección de características. Esta realimentación incrementa el coste computacional del sistema, sin embargo permite mejorar la precisión de las predicciones y alcanzar un equilibrio entre la precisión conseguida y el número de sondas seleccionadas.

## 4. Resultados de los experimentos

Para este trabajo se han utilizado tres conjuntos de datos (GDS3057 [23], GDS3268 [21] y GDS4758 [9]) obtenidos de la página del NCBI [6]. Además, se ha probado la efectividad del método propuesto comparando los resultados obtenidos con los obtenidos por el método de filtrado por ganancia de información, método ya utilizado en la literatura, como en [24].

### 4.1. Bloque1: Preprocesado de los datos

Tras este paso, los conjuntos de datos pierden todas aquellas sondas de control y las que tuviesen valores no definidos. El resto de datos es trasladado al origen de coordenadas y posteriormente transformado en base a la desviación estándar de las sondas.

### 4.2. Bloque2: Selección de características

**Filtrado 1: Prueba U de Mann-Whitney** Para realizar el test estadístico de Mann-Whitney se ha utilizado la función *wilcox.test* de *R*. En este paso se eliminan todas aquellas sondas que tengan un p-valor mayor de 0,05.

**Filtrado 2: Ganancia de información** En este segundo paso del bloque de selección de características, se realiza un ranking utilizando el método de ganancia de información y el p-valor obtenido en el paso previo. En los conjuntos de datos utilizados no se ha realizado ningún filtro, pues había menos de 2000 características tras el último paso en todos ellos. El método de ganancia de información se ha implementado con la función *information.gain* del paquete *FSelector*.

**Clustering jerárquico** El clustering se realiza podando de forma automática un árbol jerárquico, realizado con la función *agnes* del paquete *cluster* [15]. Esta poda es ejecutada por la función *cutreeDynamic* del paquete *dynamicTreeCut* [14]. Existe, en todos los conjuntos salvo el *GDS3268*, el cluster 0, el cual no es un cluster en sí, sino el conjunto de todas las sondas que no pueden ser agrupadas de alguna forma.

**Filtrado 3: Selección por correlación** Mediante la modificación de los umbrales de correlación y de información, este paso interacciona con el bloque de clasificación. El objetivo es encontrar un equilibrio entre una buena precisión de la clasificación y la utilización de un número pequeño de características. Estos valores son porcentajes dados en función del valor máximo que existe para cada tipo de umbral en el conjunto de datos. El valor de estos umbrales tras la ejecución completa del sistema inteligente, junto al número de genes obtenidos en cada dataset pueden ser observados en la tabla 1.

Dataset	Umbral de correlación (%)	Umbral de información (%)	Sondas seleccionadas
GDS3057	0.5	70	14
GDS3268	0.022	46	10
GDS4758	15	60	6

Tabla 1: Parámetros y número de sondas seleccionadas tras el tercer filtrado.

### 4.3. Bloque3: Clasificación de los datos

Para realizar la clasificación se utiliza una máquina de soporte vectorial. Para validar los resultados obtenidos de la clasificación se realizan validaciones cruzadas, de 5 folds cada una. Además, para cada uno de ellos, las muestras que se encargarán de la realización del entrenamiento del clasificador y de su posterior prueba son seleccionadas de forma aleatoria, pero sin repetir las muestras que se utilizan para las pruebas en ninguno de los 5 folds. Esto permite que, cada vez que se ejecute la clasificación se obtengan resultados diferentes, y disponer así, de más cantidad de pruebas que nos ayuden a validar los resultados de este sistema inteligente comparándolos con los del método de ganancia de información. En total se han realizado 5 clasificaciones por cada conjunto de datos, que

multiplicadas a los 5 errores obtenidos de la validación cruzada que da cada clasificación, se obtienen 25 errores de clasificación por cada conjunto de datos.

#### 4.4. Experimentación con la ganancia de información

Para poder comprobar realmente la efectividad del sistema inteligente propuesto, se debe realizar una comparación con otro método de la literatura. En este caso se opta por el método de filtrado por ganancia de información. Se ha elegido este método, porque ya ha sido implementado parcialmente al ser utilizado en uno de los pasos del sistema inteligente propuesto.

**Filtrado por la ganancia de información** Para realizar este filtrado se hace un ranking en base a la ganancia de información que tiene cada característica, y se seleccionan las 10, 20 y 30 primeras características de cada conjunto de datos, obteniendo tres nuevos conjuntos por cada uno de los anteriores.

**Clasificación** La clasificación se realiza de la misma forma que en el apartado 4.3 sólo que esta vez se utilizan los conjuntos de datos obtenidos del filtrado por ganancia de información.

#### 4.5. Comparación de resultados

Aunque en la tabla 2, se puede ver que el sistema inteligente propuesto mejora el error medio de clasificación en prácticamente todas las pruebas, es necesario realizar un análisis de varianza (por su terminología en inglés *ANalysis Of VAriance* ANOVA), entre los resultados del sistema inteligente propuesto y aquellos obtenidos por la selección de características realizada con el método de ganancia de información. Para esta prueba ANOVA [12] se ha seleccionado un p-valor de 0.1.

Dataset	Método	Error medio (%)	Sondas seleccionadas
GDS3057	GI 10	1.66	10
	GI 20	4.33	20
	GI 30	0.00	30
	Sistema propuesto	0.00	13
GDS3268	GI 10	20.92	10
	GI 20	18.05	20
	GI 30	15.07	30
	Sistema propuesto	16.00	10
GDS4758	GI 10	13.06	10
	GI 20	6.66	20
	GI 30	8.53	30
	Sistema propuesto	7.73	6

Tabla 2: Comparación del error medio y del número de sondas seleccionadas.

Dataset	Método	P-valor respecto al sistema propuesto
GDS3057	GI 10	0.01
	GI 20	0.00
	GI 30	NA
GDS3268	GI 10	0.00
	GI 20	0.21
	GI 30	0.52
GDS4758	GI 10	0.00
	GI 20	0.59
	GI 30	0.66

Tabla 3: P-valor obtenido en el análisis de la varianza.

En la tabla 3 se pueden ver los resultados obtenidos, para cada conjunto de datos, al realizar la prueba ANOVA con los resultados del sistema inteligente y los de los tres conjuntos de datos obtenidos del método de ganancia de información (GI10, GI20 y GI30).

De todo lo documentado en este apartado se deduce que, en todos los conjuntos de datos utilizados, el sistema inteligente mejora de un modo u otro los obtenidos por el método de ganancia de información. También se concluye que el método propuesto, puede encontrar un conjunto de sondas más óptimo que el método de ganancia de información, sin que el error obtenido en la clasificación empeore estadísticamente. En el caso del método GI30 para el conjunto de datos *GDS4758*, utilizar más sondas no mejoran la predicción respecto al GI20. Por esto, seleccionar un conjunto mayor de sondas, puede ser indicativo de selección de datos ruidosos.

## 5. Conclusiones y líneas de trabajo futuras

La investigación realizada en este trabajo de fin de máster ha permitido sentar las bases para el desarrollo de un sistema inteligente capaz de seleccionar un subconjunto de sondas dentro de conjuntos de datos obtenidos de *microarrays*, para predecir si una muestra de ese conjunto pertenece a una de dos clases diferentes.

Los resultados obtenidos de experimentos realizados sobre tres conjuntos de datos obtenidos de la página dle NCBI [6] (*GDS3057*, *GDS3268* y *GDS4758*), se comparan a los obtenidos de utilizar el método de ganancia de información para esos mismos conjuntos de datos. Esta comparación muestra que, con el método propuesto y de forma general, se obtienen mejores resultados, tanto en precisión en la clasificación como en número de características seleccionadas, que los del método de filtrado por ganancia de información.

Una de las ventajas de este sistema inteligente es que todos los algoritmos y métricas propuestos pueden ser sustituidos por otros que también permitan seguir la estrategia que se desarrolla. Lo que abre la puerta a experimentos futuros que permitan aceptar o rechazar dichos cambios en los algoritmos

Como continuación, se plantea trabajar en el apartado del preprocesado de los datos. Se hace necesario invertir tiempo en el estudio y desarrollo de métodos que sean eficaces para la estimación de NAs, que ayuden a mejorar la clasificación en todos aquellos conjuntos de datos que posean una considerable cantidad de estos valores perdidos.

Otro de los puntos interesantes es el tercer paso de filtrado. Se considera un punto importante, el centrar la investigación futura en cómo mejorar la eliminación de información redundante dejando sólo aquella que es indispensable. Todo esto sin que se vea perjudicada la clasificación y que el tiempo de ejecución no se dispare.

Por último, para no ser tan dependientes de un algoritmo de clasificación, en vista a la interacción que hay entre el clasificador y el último paso de filtrado, se puede aprovechar el uso de más clasificadores en conjunto que permitan reducir la dependencia de un sólo tipo de clasificador y ayuden a mejorar la precisión de la predicción.

## Referencias

1. A. Alzheimer's, "2015 alzheimer's disease facts and figures." *Alzheimer's & dementia: the journal of the Alzheimer's Association*, vol. 11, no. 3, p. 332, 2015.
2. D. ARUNA and D. NANDAKISHORE, "Hybrid ensemble gene selection algorithm for identifying biomarkers from breast cancer gene expression profiles," 2013.
3. V. Bolón-Canedo, N. Sánchez-Marono, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, "A review of microarray datasets and applied feature selection methods," *Information Sciences*, vol. 282, pp. 111–135, 2014.
4. C.-C. Chiu, S.-Y. Chan, C.-C. Wang, and W.-S. Wu, "Missing value imputation for *microarray* data: a comprehensive comparison study and a web tool," *BMC systems biology*, vol. 7, no. 6, p. S12, 2013.
5. C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
6. R. Edgar, M. Domrachev, and A. E. Lash, "Gene expression omnibus: Ncbi gene expression and hybridization array data repository," *Nucleic acids research*, vol. 30, no. 1, pp. 207–210, 2002.
7. E. Hemphill, J. Lindsay, C. Lee, I. I. Măndoiu, and C. E. Nelson, "Feature selection and classifier performance on diverse bio-logical datasets," *BMC bioinformatics*, vol. 15, no. 13, p. S4, 2014.
8. Z. M. Hira and D. F. Gillies, "A review of feature selection and feature extraction methods applied on *microarray* data," *Advances in bioinformatics*, vol. 2015, 2015.
9. M. Hokama, S. Oka, J. Leon, T. Ninomiya, H. Honda, K. Sasaki, T. Iwaki, T. Ohara, T. Sasaki, F. M. LaFerla *et al.*, "Altered expression of diabetes-related genes in alzheimer's disease brains: the hisayama study," *Cerebral cortex*, vol. 24, no. 9, pp. 2476–2488, 2013.
10. M. Jafari, B. Ghavami, and V. S. Naeini, "An information gain approach to infer gene regulatory networks," in *Biomedical Engineering (ICBME), 2015 22nd Iranian Conference on*. IEEE, 2015, pp. 81–86.
11. T. M. Khoshgoftaar, D. J. Dittman, R. Wald, and W. Awada, "A review of ensemble classification for dna microarrays data," in *Tools with Artificial Intelligence (ICTAI), 2013 IEEE 25th International Conference on*. IEEE, 2013, pp. 381–389.

12. H.-Y. Kim, "Analysis of variance (anova) comparing means of more than two groups," *Restorative dentistry & endodontics*, vol. 39, no. 1, pp. 74–77, 2014.
13. M. B. Kursa, "Robustness of random forest-based gene selection methods," *BMC bioinformatics*, vol. 15, no. 1, p. 8, 2014.
14. P. Langfelder, B. Zhang, and S. Horvath, "Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r," *Bioinformatics*, vol. 24, no. 5, pp. 719–720, 2008.
15. M. Maechler, P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik, *cluster: Cluster Analysis Basics and Extensions*, 2017, r package version 2.0.6 — For new features, see the 'Changelog' file (in the package source).
16. S. Mahajan, S. Singh *et al.*, "Review on feature selection approaches using gene expression data," *Imperial Journal of Interdisciplinary Research*, vol. 2, no. 3, 2016.
17. H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, pp. 50–60, 1947.
18. E. Mohamed, E. M. El Houby, K. T. Wassif, and A. I. Salah, "Survey on different methods for classifying gene expression using *microarray* approach," *International Journal of Computer Applications*, vol. 150, 2016.
19. A. Natarajan and T. Ravi, "A survey on gene feature selection using *microarray* data for cancer classification," *Int. J. Comput. Sci. Commun*, vol. 5, pp. 126–129, 2014.
20. T. Nguyen, A. Khosravi, D. Creighton, and S. Nahavandi, "Hierarchical gene selection and genetic fuzzy system for cancer *microarray* data classification," *PloS one*, vol. 10, no. 3, p. e0120364, 2015.
21. C. L. Noble, A. R. Abbas, J. Cornelius, C. W. Lees, G.-T. Ho, K. Toy, Z. Modrusan, N. Pal, F. Zhong, S. Chalasani *et al.*, "Regional variation in gene expression in the healthy colon is dysregulated in ulcerative colitis," *Gut*, vol. 57, no. 10, pp. 1398–1405, 2008.
22. C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.
23. D. L. Stirewalt, S. Meshinchi, K. J. Kopecky, W. Fan, E. L. Pogossova-Agadjanyan, J. H. Engel, M. R. Cronk, K. S. Dorcy, A. R. McQuary, D. Hockenbery *et al.*, "Identification of genes with abnormal expression changes in acute myeloid leukemia," *Genes, Chromosomes and Cancer*, vol. 47, no. 1, pp. 8–20, 2008.
24. B. Sui, "Information gain feature selection based on feature interactions," Ph.D. dissertation, University of Houston, 2013.
25. S. Tarek, R. A. Elwahas, and M. Shoman, "Gene expression based cancer classification," *Egyptian Informatics Journal*, 2016.
26. L. A. Torre, F. Bray, R. L. Siegel, J. Ferlay, J. Lortet-Tieulent, and A. Jemal, "Global cancer statistics, 2012," *CA: a cancer journal for clinicians*, vol. 65, no. 2, pp. 87–108, 2015.

# Sistema de Recomendación de Música

Javier Pérez-Marcos<sup>1</sup>, Vivian López Batista<sup>1</sup> y María N. Moreno García<sup>1</sup>

Departamento de Informática y Automática, Facultad de Ciencias  
Plaza de los Caídos s/n, 37008, Salamanca, España  
{jpmarcos, vivian, mmg}@usal.es

**Resumen** En los últimos años la cantidad de música que un usuario puede escuchar es enorme. La combinación de los *smartphones* junto con los servicios de *streaming* de música permite que sea posible acceder a prácticamente la totalidad de la música desde cualquier parte. Esta cantidad de contenido es inabarcable para los usuarios, por lo que los sistemas de recomendación son una solución perfecta. Sin embargo, estos sistemas plantean una serie de problemas como son *cold start*, *sparsity*, *gray sheep* y la escalabilidad. El uso de información adicional proveniente de sistemas de etiquetado social en los sistemas de recomendación puede solucionar parte de estos problemas. En este trabajo se propone un sistema de recomendación de música basado en etiquetado social que solventa los problemas planteados.

**Keywords:** sistema de recomendación, etiquetado social, folksonomía, recomendación de música

## 1. Introducción

La relación entre la música y las personas ha avanzado siempre paralela a la tecnología. Desde el gramófono en 1877 hasta los *smartphones* de hoy en día, el contenido musical que los usuarios tienen disponible se ha ido incrementando enormemente. Sin embargo, esa gran cantidad de contenido también supone un problema para los usuarios. Los sistemas de recomendación permiten a los usuarios navegar, descubrir y categorizar de manera personalizada los contenidos musicales. Sin embargo la mayoría de los sistemas de recomendación plantean una serie de inconvenientes como *cold start*, *sparsity*, *gray sheep* o la escalabilidad del propio sistema. Una de las posibles soluciones a estos problemas la podemos encontrar en los sistemas de etiquetado social. Estos sistemas permiten a los usuarios anotar de manera personalizada los recursos para categorizarlos. Como resultado de estos sistemas de etiquetado social se obtienen las folksonomías, que son una taxonomía social de los recursos fruto del conocimiento colectivo. Este conocimiento puede ser utilizado para aumentar la información de los sistemas de recomendación y poder solucionar los problemas que se han mencionado antes. En este trabajo se propone un sistema de recomendación de música basado en etiquetado social. Puesto que no todos los servicios de *streaming* de música permiten el etiquetado de sus recursos, el sistema presentado

modelara una folksonomía a partir de otros servicios de etiquetado social. A partir de la información del propio servicio de *streaming* junto con la folksonomía inducida, el sistema realizara recomendaciones y pondrá solución a alguno de los problemas planteados.

## 2. Trabajos Relacionados

A continuación se describen algunos de los trabajos más recientes sobre sistemas de recomendación basados en etiquetado social en el contexto de la música.

En el trabajo de Ness et al. [13] presentan un sistema para el etiquetado automático de canciones. El sistema que describen hace uso del análisis del contenido de audio de la música para generar las predicciones. Los autores defienden que el uso de sistemas automáticos de etiquetado son solución a los problemas de arranque en frío. Además, comentan que la generalización apilada puede ser utilizada para superar al resto de sistemas de etiquetado automático conocidos.

Zhao et al. [16] proponen un esquema para la recomendación de etiquetas mediante el uso de atributos múltiples explícitos basados en la similitud semántica de las etiquetas y el contenido de la música. Los autores aseguran que en el enfoque presentado el espacio de atributos se limita inicialmente a un conjunto que minimiza la pérdida semántica y el ruido de las etiquetas, mientras que garantiza la diversidad de atributos.

Tatli et al. [15] presentan un enfoque híbrido para la recomendación de música. A partir de un conjunto de etiquetas proporcionadas por usuarios construyen una estructura jerárquica (ontología) extraída de los géneros de nivel superior de la DBpedia<sup>1</sup>. Mediante el uso de esta estructura bien definida se consigue una reducción de la dimensionalidad en los perfiles de los usuarios y de los ítems.

La propuesta de Hariri et al. [8] presenta un modelo basado en el contexto que utiliza las secuencias de canciones para predecir el siguiente tema. Este enfoque utiliza el modelado de temas para buscar patrones dentro de las secuencias de reproducción. El modelado de dichos temas se realiza mediante Análisis Semántico Latente (LDA), de tal manera que dada una secuencia de canciones el sistema es capaz de predecir los temas de dicha secuencia.

Kaminskas et al. [10] proponen un enfoque diferente al de los anteriores autores. En este trabajo presentan un sistema de recomendación orientado al contexto, recomendando música adecuada para lugares de interés (POIs). Este enfoque híbrido combina la representación de los POIs y la música mediante etiquetas, y el conocimiento basado en las relaciones semánticas de los dos tipos de elementos. El sistema propuesto realiza una recomendación híbrida entre las recomendaciones basadas en conocimiento y las recomendaciones basadas en etiquetas.

Otro trabajo interesante es el de Fernandez-Tobías et al. [7] en donde presentan un modelo emocional basado en etiquetas sociales. Este modelo se basa en un léxico generado automáticamente que describe los sentimientos mediante el significado de los sinónimos y antónimos. Mediante este modelo, el sistema

<sup>1</sup> <http://wiki.dbpedia.org/>

es capaz de relacionar folksonomías emocionales entre dominios, como es el caso del cine y la música, a través de la creación de perfiles orientados a las emociones basados en etiquetas.

Craw et al. [4] muestran en su trabajo una aproximación híbrida que extiende la representación de las canciones basada en etiquetas mediante el conocimiento semántico extraído de las canciones similares. De esta manera, según sus autores, son capaces de lidiar con el problema *long tail*.

Por último, el trabajo de Ding et al. [6] presenta un método basado en contenido por el cual consiguen mejorar el grado de preferencia del usuario con el uso de etiquetas ponderadas. Los autores comentan en este trabajo que a diferencia del resto de métodos basados en contenido, en donde los sistemas tan solo realizan predicciones binarias de si le gustara la canción o no, su propuesta consigue aportar un grado de preferencia a la predicción.

### 3. Metodología

El sistema de recomendación de música propuesto esta basado en el etiquetado social. El enfoque que se ha tomado ha sido pensado para servicios de *streaming* música (p.e. Spotify<sup>2</sup>) que carecen del componente social. De esta manera, es posible definir una folksonomía sobre el servicio de *streaming* aunque no sea posible etiquetar los ítems. El sistema esta dividido en tres módulos:

- Módulo de creación de perfiles: En este módulo se induce la folksonomía y a partir de ella los perfiles de usuario, para posteriormente utilizarlos en los siguientes módulos.
- Módulo de creación de comunidades: En este módulo se detectan las comunidades en las que se agrupan los usuarios y los representantes de cada comunidad.
- Módulo de recomendación: Es el último módulo del sistema. Se apoya en los dos anteriores. En este módulo se realizan las predicciones del usuario activo a partir de su perfil (obtenido en el primer módulo) sobre la comunidad a la que pertenece (obtenida del segundo módulo).

#### 3.1. Módulo de creación de perfiles

En este módulo se obtiene los perfiles basados en etiquetas de los usuarios. El módulo cuenta con dos conjuntos de datos de entrada (Fig. 1), los perfiles de los ítems basados en etiquetas y los perfiles de los usuarios basados en valoraciones. A partir de dichos datos de entrada se obtienen los perfiles de los usuarios basados en etiquetas como salida. El conjunto de de perfiles de ítems se crea a partir de la folksonomía obtenida de la red o redes sociales. Los perfiles de los ítems se obtienen a través de la proyección de la folksonomía  $\mathbb{F}^*$  en la dimensión de los

<sup>2</sup> <https://www.spotify.com/>

usuarios, representados como una matriz bidimensional ítem-etiqueta. Cada uno de los valores  $r_{i,t}$  de la matriz vienen dados por la siguiente ecuación:

$$r_{i,t} = \sum_{u \in U^*} y_{u,i,t} \quad (1)$$

donde  $u \in U^*$  son los usuarios de la red social e  $y_{u,i,t} \in Y^*$  es la relación entre el usuario  $u$ , el ítem  $i$  y la etiqueta  $t$ . Por lo que el conjunto de perfiles de ítems se representa como una matriz  $M^I$  *ítems*  $\times$  *etiquetas* en donde cada fila es el perfil  $P_i^I$  de un ítem  $i \in I$ .

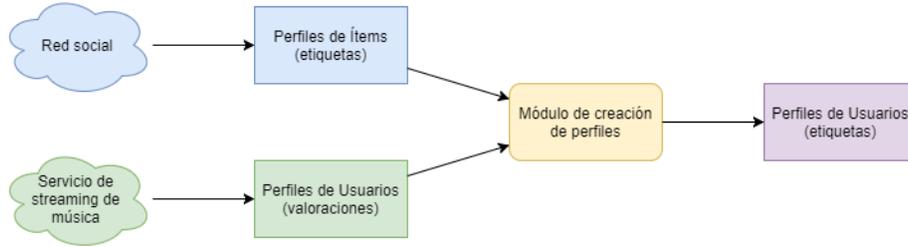


Figura 1: Módulo de creación de perfiles

El conjunto de perfiles de usuarios se obtiene a través de los servicios de *streaming*. Dicho conjunto de perfiles de usuarios se representa como una matriz  $M^U$  *usuarios*  $\times$  *ítems* en donde cada fila es el perfil  $P_u^U$  de un usuario  $u \in U$ . Ambos conjuntos son normalizados de tal manera que la *norma 1* de cada perfil sea 1, es decir, que se cumpla la siguiente restricción:

$$\|P_i^I\|_1 = \|P_u^U\|_1 = 1, \forall i \in I, u \in U \quad (2)$$

En el caso del sistema propuesto, los perfiles  $P_i^I$  se han normalizado según la frecuencia de las etiquetas dentro de dicho perfil. A partir de las matrices normalizadas  $M^I$  y  $M^U$  se induce la folksonomía  $\mathbb{F}$  del sistema (Fig. 2). Cada relación ternaria  $(u, i, t) \in Y$  se calcula mediante la siguiente ecuación:

$$y_{u,i,t} = r_{i,t} \cdot s_{u,i} | r_{i,t} \in M^I, s_{u,i} \in M^U \quad (3)$$

obteniéndose la folksonomía  $\mathbb{F} := (U, T, I, Y)$ , donde  $U$  son los usuarios del sistema,  $T$  son las etiquetas obtenidas del etiquetado social,  $I$  son los ítems del sistema e  $Y$  son las relaciones entre usuarios, etiquetas e ítems obtenidas en la Eq. 3. Al igual que para los perfiles de los ítems obtenidos del etiquetado social, la folksonomía  $\mathbb{F}$  se proyecta sobre la dimensión de los ítems, consiguiendo los perfiles de los usuarios basados en etiquetas. El conjunto de perfiles de usuario obtenido se representa como una matriz  $M$  *usuarios*  $\times$  *etiquetas* en donde cada fila es el perfil  $P_u$  de un usuario  $u \in U$ . Además, cada perfil cumple la restricción  $\|P_u\|_1 = 1$ , de tal manera que para cada perfil  $P_u$  del usuario  $u \in U$ ,  $r_{u,t}$  es la frecuencia predicha para la etiqueta  $t \in T$ .

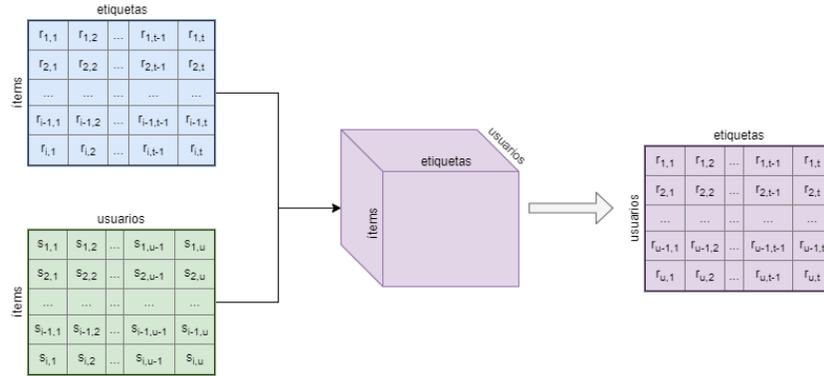


Figura 2: Inducción de la folksonomía y proyección sobre los ítems

### 3.2. Módulo de creación de comunidades

El módulo de creación de comunidades se encarga de obtener las comunidades en los que se agrupan los usuarios del sistema y los usuarios representantes de dichos *clusters* (Fig. 3). De esta forma, cuando sea necesario recomendar a un usuario no se hará sobre todos los usuarios del sistema, sino sobre los usuarios que pertenezcan a su comunidad. Además, a partir de los representantes de cada comunidad se puede determinar a que grupo pertenece un usuario novel, sin necesidad de calcular las comunidades de nuevo.

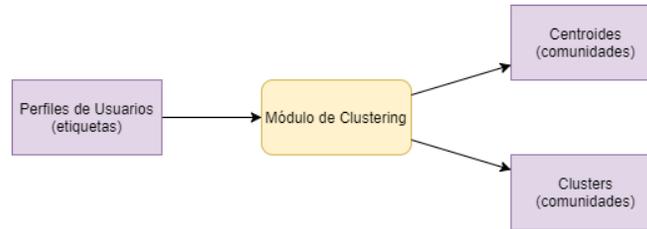


Figura 3: Módulo de *clustering*

El módulo toma como entrada la matriz de perfiles de usuarios obtenida en el módulo anterior. Sobre esta matriz se construye un grafo en el que cada nodo es un usuario y el peso de las aristas que los unen es la similitud entre ellos. Para obtener la similitud entre usuarios se ha utilizado el análisis semántico latente [11,12]. Mediante la aplicación del LSA se pasa de una matriz *usuarios – ítems*  $M$  a una matriz simétrica *usuarios – usuarios*  $M^U$  en el que cada registro se corresponde con el valor de similitud  $s_{u,v}$  del usuario  $u$  con el usuario  $v$  con  $u, v \in U$ . Dicha matriz se corresponde con la matriz de adyacencia del grafo. Al aplicarse el coeficiente de correlación de Pearson como medida

de similitud, aquellos pesos que sean negativos son sustituidos por 0, es decir,  $s_{u,v} = 0, \forall s_{u,v} \in M^U | s_{u,v} < 0$ . Al ser un grafo no dirigido sobre una matriz de adyacencia simétrica, solo se tendrán en cuenta los registros  $s_{u,v}$  tales que  $u > v$ .

La obtención de las comunidades y los usuarios que pertenecen a ellas (Fig. 4) se realiza aplicando el método de Louvain para la detección de comunidades [1]. De las comunidades obtenidas se calculará la puntuación *PageRank* [14,2] para cada nodo de dicha comunidad. Aquellos nodos con la puntuación más alta de cada comunidad se seleccionaran como representante de dicha comunidad.

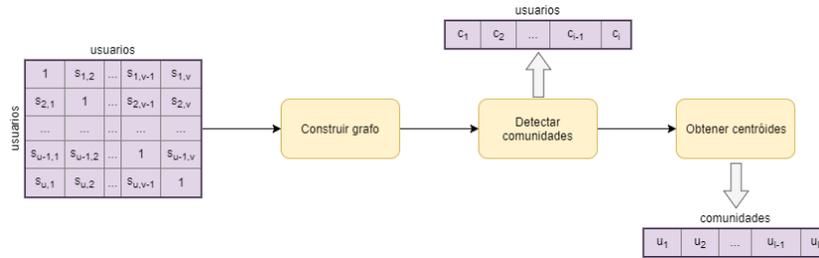


Figura 4: Obtención de comunidades y representantes

### 3.3. Módulo de recomendación

El módulo de recomendación es el encargado de realizar las predicciones de ítems al usuario activo (Fig. 5). La metodología elegida para realizar dichas predicciones es el filtrado colaborativo (CF). El CF utiliza las valoraciones sobre ítems pasadas de los usuarios para realizar recomendaciones. En el caso del CF basado en etiquetado social, se utilizan las valoraciones de los usuarios sobre las etiquetas para obtener los ítems recomendados (Fig. 6).

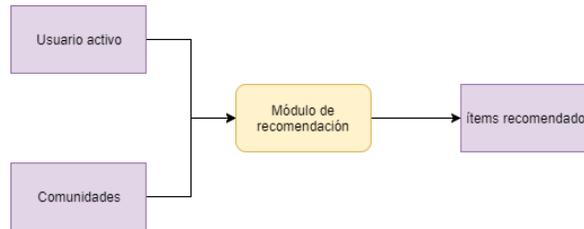


Figura 5: Módulo de recomendación

El algoritmo elegido para realizar el CF es el *k-nearest neighbors* (KNN) o los k-vecinos más cercanos. El objetivo del algoritmo es encontrar k-usuarios

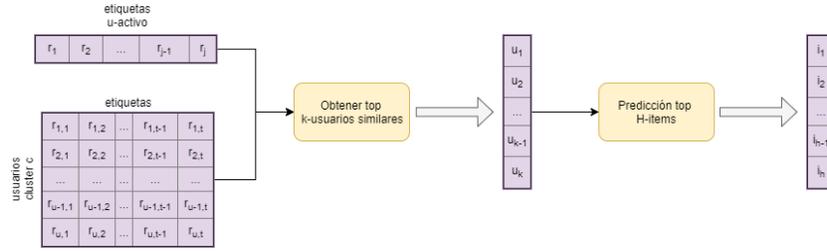


Figura 6: Predicción de los n-primeros ítems

que compartan valoraciones con el usuario activo, mediante la secuencia de los siguientes pasos:

1. Asignar un peso a todos los usuarios respecto a su similitud con el usuario activo mediante la aplicación de alguna medida de similitud.
2. Seleccionar  $k$  usuarios con el mayor coeficiente de similitud con el usuario activo, denominados vecinos.
3. Calcular la predicción a partir de la combinación ponderada de las calificaciones del subconjunto seleccionado.

Para determinar el conjunto de vecinos se utiliza la indexación semántica latente (LSI) [5]. Una vez obtenido el subconjunto de k-usuarios más similares al usuario activo, resta predecir los ítems por los que puede estar interesado. El conjunto de ítems recomendados se obtiene mediante la media ponderada de las desviaciones de la media de los vecinos:

$$P_{a,i} = \bar{s}_a + \frac{\sum_{u \in K} (s_{u,i} - \bar{s}_u) \times w_{a,u}}{\sum_{u \in K} w_{a,u}} \quad (4)$$

donde  $P_{a,i}$  es la predicción para usuario activo  $a$  sobre el ítem  $i$ ,  $w_{a,u}$  es la similitud entre el usuario activo y el usuario  $u$ , y  $K$  es el subconjunto de k-usuarios más similares.

Uno de los problemas del CF es la escalabilidad, a medida que crece el número de usuarios crece el tiempo de cálculo. Por tanto, para hacer el sistema más escalable, se hace uso de las comunidades y representantes del módulo anterior para determinar la comunidad de usuario activo y realizar las predicciones solo sobre dicha comunidad (Fig. 7). Los pasos a seguir son los siguientes:

1. Calcular la similitud mediante el coeficiente coseno del usuario activo con los representantes de cada comunidad.
2. Seleccionar la comunidad del representante con mayor coeficiente de similitud.
3. Realizar CF sobre la comunidad seleccionada.

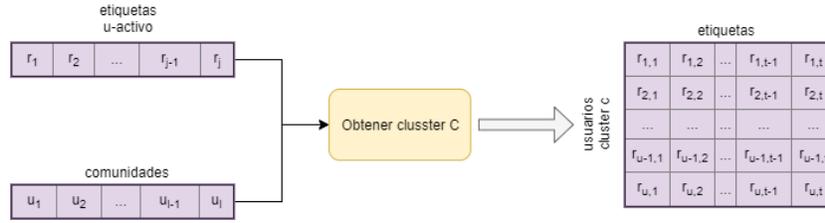


Figura 7: Determinar la comunidad del usuario activo

## 4. Experimentos y resultados

En este apartado se tratarán los experimentos realizados sobre el sistema propuesto. Primero se describirán los datos utilizados. A continuación se detallarán las métricas utilizadas para evaluar los experimentos. Por último se mostrarán los resultados obtenidos junto con un análisis crítico de los mismos. Las pruebas que se realizarán son las siguientes:

- Evaluar los perfiles obtenidos mediante el sistema propuesto con respecto a los perfiles originales.
- Evaluar e las relaciones entre los perfiles inducidos.
- Evaluar la capacidad de predicción de artistas del sistema comparándolo con otros dos métodos como son la recomendación de los artistas más populares, y el filtrado colaborativo usuario-usuario tradicional.

El conjunto de datos elegido para realizar las pruebas es *hetrec2011-lastfm-2k* [3] perteneciente al segundo taller internacional sobre Heterogeneidad de la Información y Fusión en Sistemas de Recomendación (HetRec 2011<sup>3</sup>). Los datos contienen información social de los usuarios, y de etiquetado y reproducciones de artistas de un conjunto de casi 2.000 usuarios pertenecientes a Last.fm<sup>4</sup>.

### 4.1. Métricas

Las métricas utilizadas para evaluar los experimentos son la precisión, la exhaustividad y la medida F1:

- Precisión: Es la relación entre el número de ítems relevantes predichos y el total de ítems predichos. Se define formalmente como:

$$P = \frac{(Items\ predichos) \cap (Items\ relevantes)}{(Items\ predichos)} \quad (5)$$

- Exhaustividad: Es la relación entre el número de ítems relevantes predichos y el total de ítems relevantes. Se define formalmente como:

$$R = \frac{(Items\ predichos) \cap (Items\ relevantes)}{(Items\ relevantes)} \quad (6)$$

<sup>3</sup> <http://ir.ii.uam.es/hetrec2011>

<sup>4</sup> <https://www.last.fm/>

- Medida F1: Es una medida de la exactitud de una prueba. Esta medida tiene en cuenta tanto la precisión como la exhaustividad. Se define formalmente como:

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R} \quad (7)$$

#### 4.2. Resultados y discusión

**Evaluación de los perfiles inducidos** En la Fig. 9 se puede ver el mapa de calor para los perfiles obtenidos mediante el método propuesto. Si comparamos esta gráfica con la de los perfiles reales (Fig. 8) de los usuarios se pueden sacar las siguientes conclusiones:

- No hay escasez de datos. A diferencia de los perfiles reales de los usuarios, en donde la mayoría de los usuarios utilizaban muy pocas etiquetas, en los perfiles inducidos todos los perfiles de usuarios contienen la mayoría de las etiquetas. Por lo tanto, con el método propuesto se solventa el problema de la escasez de datos.
- Se mantienen las características del perfil original. Las dos líneas características del mapa de calor de los perfiles originales se mantienen. La línea horizontal es más fácil de identificar, y se corresponde con el conjunto de etiquetas más utilizadas. La línea diagonal correspondiente a las etiquetas características de los usuarios está más difuminada, pero se sigue identificando. Por lo tanto, no se ha perdido información.

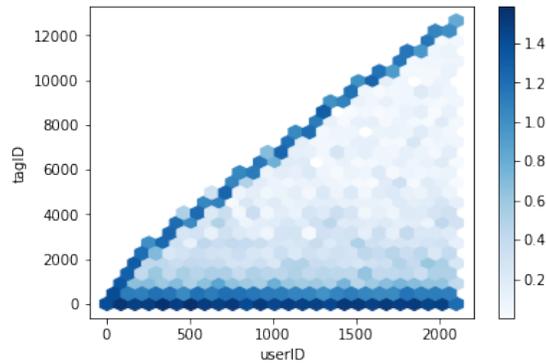


Figura 8: Mapa de calor Usuario-Etiqueta (frecuencia)

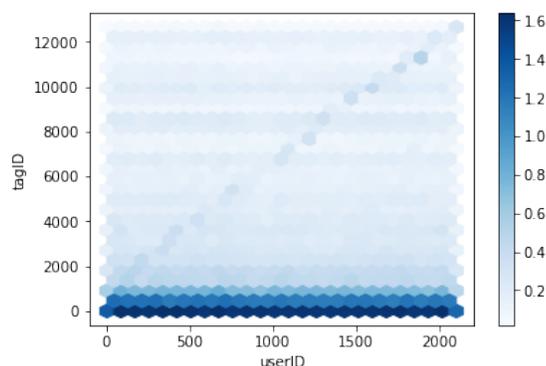


Figura 9: Mapa de calor Usuario-Etiqueta inducido (frecuencia)

**Evaluación de las relaciones entre los perfiles inducidos** En la Fig. 10 se puede ver una representación (algoritmo *Force Atlas 2* [9]) de las relaciones entre los perfiles obtenidos de los usuarios, las comunidades detectadas y la importancia de los usuarios dentro de sus comunidades. De 1892 usuarios que el conjunto de datos contiene, tan solo 1708 se encuentran en el grafo (90.27% de los usuarios), con un total de 951421 relaciones (densidad del grafo de 0.653). Los 184 restantes (9.73% de los usuarios) son usuarios que no poseen ninguna similitud con el resto de usuarios. Por lo tanto, estos usuarios podrían ser catalogados como ovejas negras. Dentro del grafo se han detectado 6 comunidades distribuidas tal y como se puede ver en la Tab. 1. Por el tamaño de los nodos se representa el valor de *PageRank* obtenido dentro del sub-grafo.

Comunidad	% de Usuarios
Naranja	21.95
Azul	17.95
Verde claro	17.71
Roja	12.57
Morada	8.81
Verde oscuro	2.33

Tabla 1: Distribución de los usuarios en las comunidades

**Evaluación del sistema propuesto** La evaluación del sistema propuesto se ha hecho en dos pruebas. En la primera prueba, denominada *prueba general*, se evalúa la calidad de predicción general. En este experimento se utiliza, de los usuarios de prueba, dos tercios de los artistas escuchados para crear el perfil y el resto como artistas a predecir. En la segunda prueba, denominada *prueba cold*

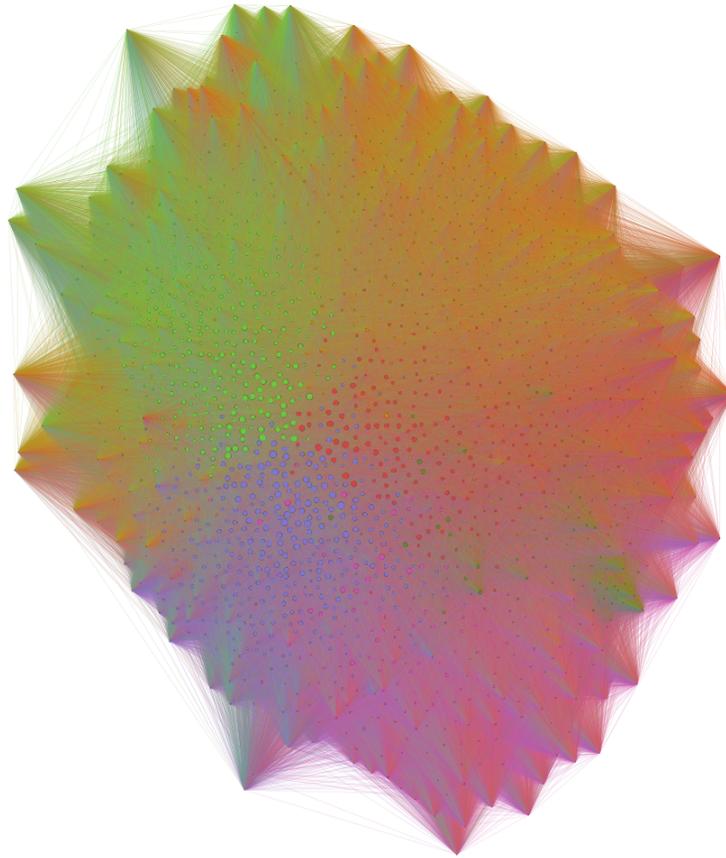


Figura 10: Grafo de relación entre usuarios

*start*, se mide la respuesta del enfoque al arranque en frío de los usuarios. Por tanto solo se utilizará un artista de los escuchados para el perfil y el resto serán los artistas a predecir. Los enfoques evaluados son la recomendación de artistas más populares, el CF basado en usuarios, el método propuesto sin comunidades y el método propuesto con comunidades.

Los resultados de la primera prueba, que muestran la precisión de cada uno de los enfoques, se pueden ver en la Fig. 11. Como se puede apreciar, el sistema propuesto sin comunidades supera ligeramente al CF basado en usuarios, tendiendo a obtener resultados similares al aumentar el *cut-off* (punto de corte) de artistas predichos. Sin embargo, el método propuesto con comunidades supera holgadamente al resto de enfoques, con similar tendencia a la del CF. El método basado en popularidad queda bastante por debajo del resto, manteniendo la precisión pese al *cut-off*. En la Tab. 2 se muestra la medida F1 de cada prueba

para la predicción de 10 Artistas. Se puede ver que el método propuesto con comunidades es el que mejor puntuación tiene, seguido por el método propuesto sin comunidades, el CF y el basado en popularidad, en ese orden.

En cuanto a la prueba *cold start*, sus resultados en cuanto a precisión no distan de la prueba anterior. CF y el método propuesto sin comunidades obtienen resultados similares, con el método propuesto significativamente por encima. El mejor enfoque es el método propuesto con el uso de comunidades, aunque a medida que aumenta el *cut-off*, sus resultados se aproximan a los de los dos enfoques anteriores. El método basado en popularidad sigue siendo el peor. En la Tab. 2 se puede ver que el método con mejor puntuación F1 es el propuesto, aunque comparado con la prueba anterior ha empeorado, es decir, sigue siendo mejor de los métodos evaluados, pero predice peor que en la prueba anterior (para el *cut off* de 10 artistas). El resto están en el mismo orden que en la anterior prueba, aunque con mejores resultados.

Método	F1@10 (p. general)	F1@10 (p. <i>cold start</i> )
Popularidad	0.0629	0.0764
CF	0.0695	0.0885
S. propuesto	0.0780	0.0954
S. propuesto (comunidades)	<b>0.1492</b>	<b>0.1036</b>

Tabla 2: Medida F1

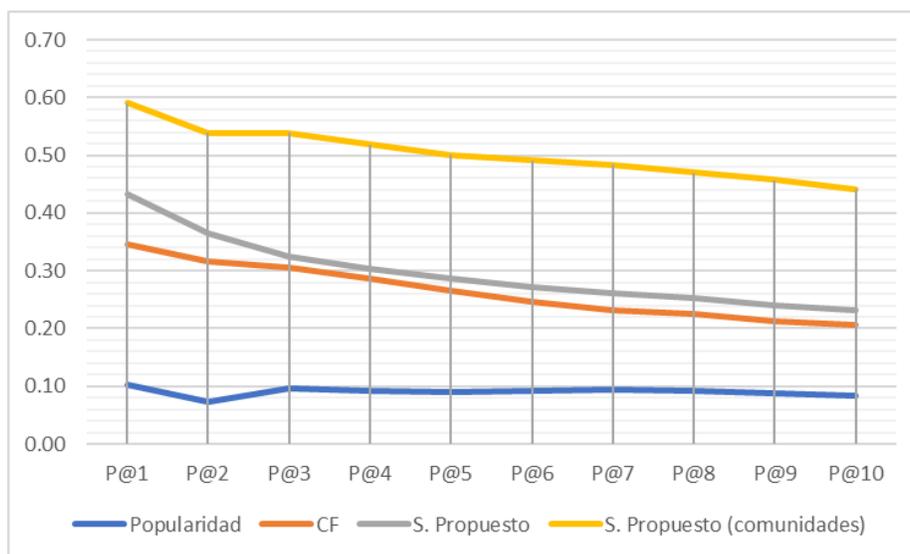


Figura 11: Gráfica precisión - Prueba general

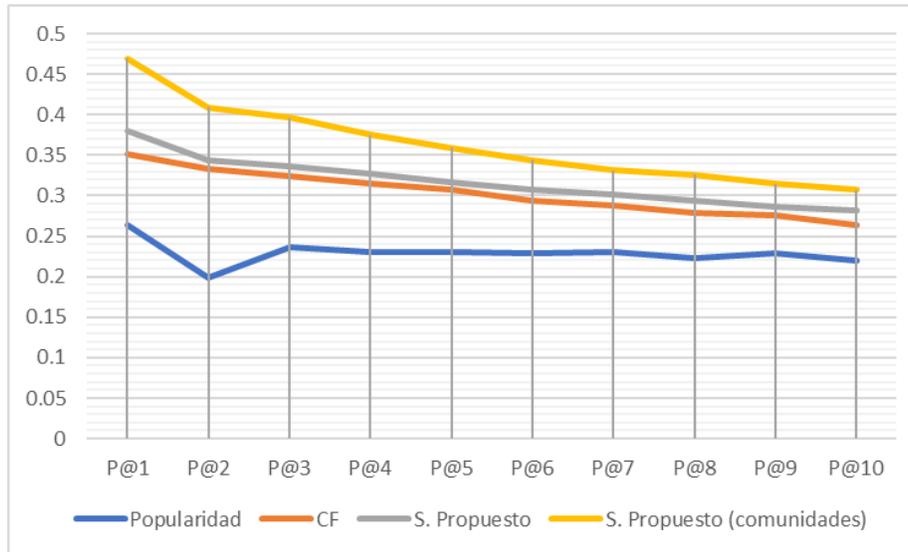


Figura 12: Gráfica precisión - Prueba *cold start*

## 5. Conclusiones y Líneas de Trabajo Futuras

El sistema de recomendación de música propuesto resuelve problemas como el arranque en frío, la escasez de datos y la detección de ovejas negras gracias al uso del etiquetado social y de las técnicas de recuperación de la información. Además utilizando métodos basados en redes sociales, como la recomendación de comunidades, se ha conseguido que el sistema sea escalable y que mejore en la predicción.

Sin embargo siguen existiendo algunos problemas, como por ejemplo que puedan aparecer canciones o artistas que todavía no hayan sido etiquetados. También hay que tener en cuenta problemas que el uso del etiquetado social ha introducido, como es el caso del *hacking* o etiquetado malintencionado.

Como líneas de trabajo futuras se podría plantear la inclusión al sistema de métodos basados en contenido que permitan predecir etiquetas para canciones o artistas noveles. También sería interesante utilizar métodos basados en contexto, que son una solución bastante buena para el problema del arranque en frío. Por último, el uso algoritmos procedentes de la inteligencia computacional, tales como *anto colony* o *particle swarm*, permitirían obtener caminos óptimos dentro de los grafos de usuarios o ítems que podrían ser utilizados para realizar predicciones.

## Referencias

1. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics: theory and*

- experiment*, vol. 2008, no. 10, p. P10008, 2008.
2. S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," *Computer networks*, vol. 56, no. 18, pp. 3825–3833, 2012.
  3. I. Cantador, P. Brusilovsky, and T. Kuffik, "2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011)," in *Proceedings of the 5th ACM conference on Recommender systems*, ser. RecSys 2011. New York, NY, USA: ACM, 2011.
  4. S. Craw, B. Horsburgh, and S. Massie, "Music recommendation: Audio neighbourhoods to discover music in the long tail," in *International Conference on Case-Based Reasoning*. Springer, 2015, pp. 73–87.
  5. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, p. 391, 1990.
  6. L. Ding, N. Zheng, J. Xu, and M. Xu, "An improved content-based music recommending method with weighted tags," in *International Conference on Multimedia Modeling*. Springer, 2015, pp. 360–371.
  7. I. Fernández-Tobías, I. Cantador, and L. Plaza, "An emotion dimensional model based on social tags: Crossing folksonomies and enhancing recommendations," in *International Conference on Electronic Commerce and Web Technologies*. Springer, 2013, pp. 88–100.
  8. N. Hariri, B. Mobasher, and R. Burke, "Context-aware music recommendation based on latent topic sequential patterns," in *Proceedings of the sixth ACM conference on Recommender systems*. ACM, 2012, pp. 131–138.
  9. M. Jacomy, T. Venturini, S. Heymann, and M. Bastian, "Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software," *PLoS one*, vol. 9, no. 6, p. e98679, 2014.
  10. M. Kaminskis, F. Ricci, and M. Schedl, "Location-aware music recommendation using auto-tagging and hybrid matching," in *Proceedings of the 7th ACM conference on Recommender systems*. ACM, 2013, pp. 17–24.
  11. T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse processes*, vol. 25, no. 2-3, pp. 259–284, 1998.
  12. C. D. Manning, P. Raghavan, and H. Schütze, "Matrix decompositions and latent semantic indexing," in *Introduction to information retrieval*. Cambridge University Press, 2008, ch. 18, pp. 403–419. [Online]. Available: <https://nlp.stanford.edu/IR-book/>
  13. S. R. Ness, A. Theocharis, G. Tzanetakis, and L. G. Martins, "Improving automatic music tag annotation using stacked generalization of probabilistic svm outputs," in *Proceedings of the 17th ACM international conference on Multimedia*. ACM, 2009, pp. 705–708.
  14. L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.
  15. I. Tatli and A. Birturk, "A tag-based hybrid music recommendation system using semantic relations and multi-domain information," in *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 548–554.
  16. Z. Zhao, X. Wang, Q. Xiang, A. M. Sarroff, Z. Li, and Y. Wang, "Large-scale music tag recommendation with explicit multiple attributes," in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 401–410.

# Detección de Objetos para Coches Autónomos

Nicolás Ugrinovic Kehdy<sup>1</sup>, William Raveane<sup>1</sup> y Angélica González Arrieta<sup>1</sup>

Departamento de Informática y Automática, Facultad de Ciencias  
Plaza de los Caídos s/n, 37008, Salamanca, España  
{nugrinovic, raveane, angelica}@usal.es

**Resumen** Este trabajo se centra en el estudio de sistemas actuales de detección de objetos que permitan su funcionamiento en tiempo real al ser aplicados en base de datos de visión artificial para coches autónomos. Se aplican y estudian diferentes técnicas de entrenamiento del sistema. Se evalúan los sistemas obtenidos basados en redes neuronales profundas con un set de datos de evaluación también de coches autónomos. Finalmente, se presentan y comparan los resultados experimentales.

**Keywords:** Detección de objetos, coches autónomos, Deep Learning, redes neuronales, tiempo real

## 1. Introducción

Este trabajo se ha centrado en el estudio de sistemas actuales de detección de objetos que permitan su funcionamiento en tiempo real al ser aplicados en base de datos de visión artificial para coches autónomos. Se utilizan dos bases de datos recopilados por una empresa de educación estrechamente relacionada con la industria de los coches autónomos<sup>1</sup>, sobre las cuales no existen trabajos publicados en la literatura. Se aplican y estudian diferentes técnicas de entrenamiento del sistema tales como *transfer learning* y *fine-tuning*. Se evalúan los sistemas obtenidos basados en redes neuronales profundas con un set de datos de evaluación también de coches autónomos. Finalmente, se presentan y comparan los resultados experimentales.

En la Sec. 2 se introduce la noción de coche autónomo y el estado actual de la tecnología relacionada. En la misma sección hace un repaso sobre los principales sistemas propuestos en la literatura para el problema de detección de objetos en tiempo real. La propuesta escogida para su posterior estudio se describe de forma teórica en la Sec. 3, donde además se presentan conceptos teóricos relacionados necesarios para su entendimiento. En la Sec. 4 se presentan los resultados de cuatro experimentos que comparan el desempeño del sistema propuesto entrenado con diferentes técnicas existentes para la tarea de detección de objetos en un entorno de coches autónomos. Finalmente, en la Sec. 5 se presentan las conclusiones finales de este trabajo y se demarcan las líneas de trabajo futuro.

---

<sup>1</sup> Web oficial: <https://www.udacity.com/>

## 2. Estado del Arte

En esta sección se introduce al problema de los coches autónomos y se repasan los principales enfoques que existen en la literatura sobre sistemas de detección de objetos, profundizando en aquellos que se consideran de tiempo real.

Las primeras demostraciones de conducción autónoma fueron iniciadas por Dickmanns y Mysliwetz en la década de los 80s. Años después, se cumplió un gran hito en el concurso de DARPA<sup>2</sup> en 2005, donde el coche autónomo bautizado como Stanley de la Universidad de Stanford fue el primero en completar el gran reto de moverse 212 kilómetros por caminos de tierra sin conductor. Desde entonces se han visto grandes avances en el campo de la conducción autónoma. En 2009 Google comenzó un proyecto para desarrollar un coche autónomo y en 2016 completó la conducción sin asistencia de un conductor humano por 1,498,000 millas. En 2015, la compañía de coches eléctricos Tesla, lanzó al mercado su producto de asistencia para conducción denominada Autopilot<sup>3</sup>.

Sin embargo, la navegación completamente autónoma en ambientes complejos no es todavía una realidad y existen grandes problemas a ser resueltos. Según [4] son dos los problemas a resolver: el primero, la inexistencia de inteligencia artificial capaz de generalizar en situaciones impredecibles y actuar a tiempo y, el segundo, contar con sistemas de percepción lo suficientemente precisos con costo accesible. Por lo tanto, se requieren sistemas precisos que puedan actuar rápidamente a tiempo para evitar comprometer la seguridad de las personas. El presente trabajo se basa en explorar el sistema de visión más adecuado capaz de resolver parte del problema de percepción: la detección de objetos en tiempo real.

Las aplicaciones en las cuales es útil la detección de objetos como ser los coches autónomos requieren que estos sistemas procesen las imágenes a mayores velocidades, logrando un funcionamiento en tiempo real (procesamiento en velocidades de 30fps o más altas). Existen, principalmente, dos modelos que se destacan por permitir la detección de objetos en tiempo real presentados por los autores de SSD [6] y YOLO [8], cada cual cuenta con una versión original y versiones mejoradas.

El sistema presentado en [8] consiste en una arquitectura unificada que permite una detección bastante rápida, 45fps en su versión original y a 155fps con una versión que utiliza una red más pequeña. Ambos casos tienen un buen desempeño alcanzando mAP (*Mean Average Precision* o la media de la precisión promedio) que se acerca al de sus contrapartes más lentas. A comparación de otros métodos [8] tiene más errores de localización, pero presenta menos falsos positivos debidos a imágenes de fondo. Sus autores sostienen es posible aprender un modelo bastante general en base de los datos, lo que le permite pasar de imágenes

<sup>2</sup> Agencia de Proyectos de Investigación Avanzados de Defensa de los EEUU.

<sup>3</sup> Web oficial: <https://www.tesla.com/autopilot>

naturales a figuras artificiales obteniendo mejores resultados frente a DPM<sup>4</sup> y R-CNN.

Al igual que los métodos presentados en los trabajos [8] y [9] el que proponen los autores de [6] utiliza una sola red neuronal profunda para la detección de objetos. Este enfoque discretiza el espacio de posibles cajas delimitadoras en un conjunto de cajas con diferentes proporciones entre sus lados y en diferentes escalas. En la predicción, el sistema realiza una clasificación para cada una de las cajas dentro del conjunto para generar los puntajes de cada clase y mejorar la localización. Adicionalmente, la red propuesta combina predicciones de varios mapas de características con diferentes resoluciones lo que mejora los resultados para objetos en diferentes escalas. Este método elimina la generación de propuesta de regiones y otras etapas utilizadas por métodos anteriores. SSD permite obtener una precisión de 74.3% de mAP a una velocidad de 43 fps en su versión SSD300 al utilizar un tamaño de lote (*batch size*) de 1. Esta red ha sido entrenada y evaluada con las bases de datos PASCAL VOC[3], COCO[5] y ILSVRC[10].

### 3. Metodología

En esta sección se describe tanto el método seleccionado para la detección de tiempo real para objetos pertinentes al entorno de un coche autónomo como las técnicas utilizadas para entrenar y evaluar el sistema. Los aspectos más importantes del trabajo realizado son: (1) se seleccionó un método de detección de tiempo real basado en CNN: YOLO9000, (2) para el entrenamiento del sistema se utiliza la base de datos [2], (3) para el entrenamiento se utilizan dos diferentes técnicas: entrenamiento de la red con inicialización de pesos aleatoria y transferencia de aprendizaje o *transfer learning* utilizando una red pre-entrenada tanto para realizar un ajuste fino (*fine tuning*) como para seguir entrenando a la red desde un punto de partida (*check-point*).

**YOLOv2.** YOLO se diferencia de sus contrapartes especialmente porque maneja el problema de detección desde un enfoque de regresión. La salida del sistema es un único tensor en donde se encuentra toda la información pertinente a la detección: la ubicación de las cajas delimitadoras y la probabilidad por clase de cada uno de los objetos dentro de las cajas. Tal como ya se señaló anteriormente, la detección se realiza con una sola red neuronal, ésta utiliza características de toda la imagen para predecir cada una de las cajas delimitadoras. Adicionalmente, se predicen las cajas delimitadoras para todas las clases dentro de la imagen simultáneamente. Por ello es que se dice que la red razona globalmente sobre la imagen para la detección.

<sup>4</sup> *Deformable parts model*, por sus siglas en inglés. Es un método que precede a las CNN que es aplicado a la detección de objetos

El sistema divide a la imagen de entrada en una cuadrícula de  $S \times S$ . Si el centro de un objeto cae sobre uno de los cuadros, entonces éste es responsable de la detección de dicho objeto. Cada una de las celdas de la cuadrícula predice  $B$  cantidad de cajas delimitadoras (*bounding boxes*) y puntajes de confianza para cada caja. Estos puntajes reflejan que tanta confianza tienen el modelo que exista el objeto dentro de la caja y qué tan precisa piensa que está ubicada la caja. Formalmente, se define a esta confianza de la siguiente manera:

$$Pr(\text{Objeto}) * IOU \quad (1)$$

Si no existen un objeto dentro de una caja, el valor de la confianza debería ser cero. En otro caso, se desea que este puntaje sea igual a la intersección sobre unión o, *Intersection Over Union*, IOU, entre la caja predicha y la anotación (ubicación real del objeto). Cada caja delimitadora contiene cinco predicciones:  $x, y, w, h$ , y la confianza. Las coordenadas  $(x, y)$  indican la ubicación del centro del BB dentro de la celda. El ancho y alto de la BB  $(w, h)$  son relativas al tamaño total de la imagen. Finalmente, la confianza indica el IOU entre la anotación y la BB predicha.

Cada celda predice  $C$  probabilidades condicionales de clases. Estas probabilidades están condicionadas en cada celda que contiene un objeto. Solo se predice un conjunto de probabilidad de clases por celda independientemente de la cantidad de cajas  $B$  que se utilizan en el modelo. Finalmente, al multiplicar estos valores se obtiene la probabilidad por clase dentro de cada caja delimitadora:

$$Pr(\text{Clase}_i | \text{Objeto}) * Pr(\text{Objeto}) * IOU = Pr(\text{Clase}_i) * IOU \quad (2)$$

En la Fig. 1 se aprecia el proceso que desarrolla YOLO para la generación de cajas delimitadoras. Primero, se divide la imagen en varias celdas, se generan las cajas delimitadoras con sus respectivos niveles de confianza. A la vez, se obtiene un mapa de probabilidades por clases para que al combinarlo con las cajas y las confianzas se obtenga el resultado final. A la salida de la red se obtiene un tensor de dimensiones  $SxSx(B * 5 + C)$ .

**Arquitectura de Red.** La arquitectura de YOLO se muestra en la Fig. 2. Las capas convolucionales iniciales se encargan de extraer características de la imagen mientras que las capas totalmente conectadas predicen las probabilidades de salida y las coordenadas. La arquitectura red se inspira en GoogleNet[11], sin embargo, en lugar de utilizar el módulo de *inception* se utiliza una capa convolucional de  $1 \times 1$ . La arquitectura original se compone de 24 capas convolucionales seguidas por dos capas totalmente conectadas.

Adicionalmente, a la versión original de YOLO existen una versión denominada *fast YOLO* que se basa en una arquitectura reducida con menor cantidad de capas convolucionales (9 en vez de 24) y menor cantidad de filtros en cada capa. Los demás parámetros no cambian de una versión a la otra. La ventaja de esta versión rápida es que al utilizar una arquitectura más reducida es más rápido tanto el entrenamiento como la detección. La velocidad de detección es más de dos veces mayor al de la versión original, sin embargo, se pierde precisión. La

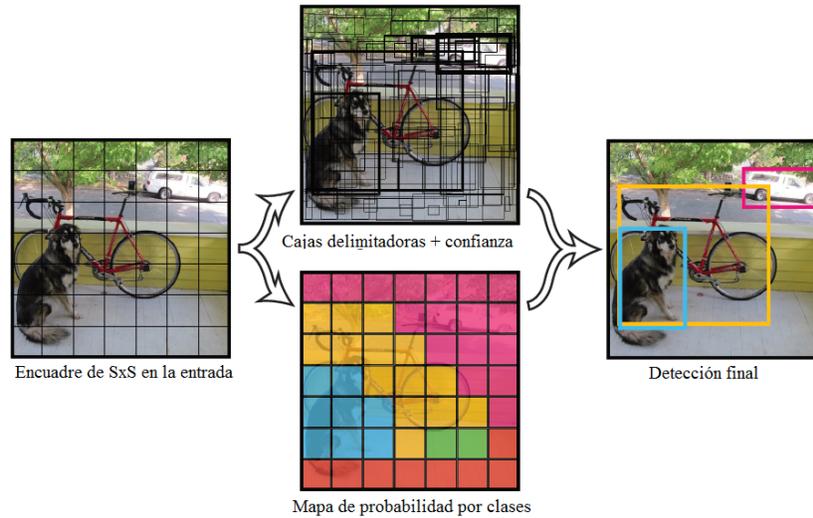


Figura1: Procedimiento de generación de BB y clasificación en YOLO (fuente:[8]).

precisión documentada es de 57.1 mAP y una velocidad de 207 fps al entrenar el modelo con la base de datos PASCAL VOC[3]. La arquitectura de la versión rápida de YOLO se muestra en la Fig. 3.

YOLOv2 se enfoca en mejorar el desempeño del sistema YOLO utilizando no una red más amplia sino más bien simplificando la arquitectura existente. En este sentido, la arquitectura resultante se compone de 19 capas convolucionales y 5 capas de *max pooling*<sup>5</sup>. Además del tamaño de la red, los cambios que se introducen en esta versión son los siguientes:

- Se utiliza *average pooling* global.
- Se utilizan filtros de 1x1 entre convoluciones de 3x3 para comprimir la representación de características en cada capa.
- Tal como lo hace la red VGG, se utilizan principalmente filtros de 3x3 y se dobla la cantidad de filtros después de cada capa de *pooling*.
- Se utiliza normalización por lotes o *batch normalization* para estabilizar el entrenamiento, acelerar la convergencia y regularizar el modelo.

El modelo resultante es denominado *Darknet-19* y su arquitectura se resume en la Tab. 1.

**Función de Pérdida.** Un aspecto sumamente importante para asegurar el correcto entrenamiento de una red neuronal es la función de pérdida. Esta función

<sup>5</sup> Técnica utilizada en Deep Learning para reducir la dimensión de los mapas de características



Tipo	Filtros	Tamaño/Stride	Salida
Convolutional	32	3 × 3	224 × 224
Maxpool		2 × 2/2	112 × 112
Convolutional	64	3 × 3	112 × 112
Maxpool		2 × 2/2	56 × 56
Convolutional	128	3 × 3	56 × 56
Convolutional	64	1 × 1	56 × 56
Convolutional	128	3 × 3	56 × 56
Maxpool		2 × 2/2	28 × 28
Convolutional	256	3 × 3	28 × 28
Convolutional	128	1 × 1	28 × 28
Convolutional	256	3 × 3	28 × 28
Maxpool		2 × 2/2	14 × 14
Convolutional	512	3 × 3	14 × 14
Convolutional	256	1 × 1	14 × 14
Convolutional	512	3 × 3	14 × 14
Convolutional	256	1 × 1	14 × 14
Convolutional	512	3 × 3	14 × 14
Maxpool		2 × 2/2	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	512	1 × 1	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	512	1 × 1	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	1000	1 × 1	7 × 7
Avgpool		Global	1000
Softmax			

Tabla 1: Arquitectura de red YOLOv2. Tabla extraída de[9]

$$\begin{aligned}
Loss = & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[ (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
& + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[ \left( \sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left( \sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} (C_i - \hat{C}_i)^2 \\
& + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} (C_i - \hat{C}_i)^2 \\
& + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2 \quad (3)
\end{aligned}$$

Esta función de pérdida optimiza el error cuadrático medio a la salida del modelo, donde se toma en cuenta tanto el error de predicción de coordenadas como de clasificación. Se utiliza esta función por simplicidad, sin embargo, si no se toman en cuenta los parámetros  $\lambda_{\text{coord}}$  y  $\lambda_{\text{noobj}}$  que asignan pesos a cada error por separado, el sistema tiende a volverse inestable a la hora de ser entrenado. Esta inestabilidad se debe a una gran diferencia entre gradientes en celdas que contienen objetos frente a aquellas que no los contienen.

### 3.1. Transfer Learning

En la práctica, no es común entrenar una CNN desde el inicio (con inicialización aleatoria de pesos) cuando no se cuenta con una base de datos adecuada y lo suficientemente grande. Por el contrario, lo que se suele hacer es utilizar una CNN previamente entrenada sobre una base de datos extensa (por ejemplo, ImageNet, que contiene 1.2 millones con 1000 categorías). Luego se utiliza esta red pre-entrenada ya sea como un extractor de características como etapa inicial de un sistema más grande o como un punto de partida para entrenar la misma red con leves modificaciones en las capas de salida con una base de datos diferente y específica a la aplicación o problema que se plantea resolver. De esta forma existen tres principales escenarios donde se puede aplicar el concepto de *transfer learning*:

- **CNN como extractor de características:** Consiste en quitar la última capa totalmente conectada de la CNN pre-entrenada con una base de datos extensa y conectar la red a un clasificador, que normalmente, es otra capa totalmente conectada que se adecua al nuevo dataset que se quiere tratar. De esta forma, la CNN actúa como un extractor de características como paso previo a la clasificación, en este caso se hace un entrenamiento solo para ajustar la etapa de clasificación mientras que el resto de la red queda intacto.
- **Ajuste fino de la CNN:** La segunda estrategia consiste en no solo entrenar a la última capa de clasificación sino que hacerlo con toda la red permitiendo realizar la retro-propagación para ajustar todos los pesos. En algunos casos se congelan los parámetros de las primeras capas convolucionales haciendo que, explícitamente, la retro-propagación no tenga efecto en ellos. Este último procedimiento se lo toma en cuenta para evitar el sobre-entrenamiento (*overfitting*) dado que estas capas actúan como extractores de características generales que son útiles para muchas tareas. En contraposición, las últimas capas de la red se vuelven, progresivamente, extractores de características más específicas de la base de datos con la que fue entrenada.
- **Punto de partida (*check point*) de modelos pre-entrenados:** Dado que las CNN pueden tardar mucho tiempo en ser entrenadas, alrededor de 2 o 3 semanas dependiendo de la base de datos, muchas personas entrenan las redes y hacen públicos los parámetros conseguidos en este proceso para que otros se beneficien de ello. Con un punto de partida es posible hacer un ajuste fino de la red.

De entre estas tres estrategias, para este trabajo se escogen las últimas dos. La primera, el ajuste fino de la red YOLOv2 en su versión reducida (más rápida) pre-entrenada con la base de datos PASCAL VOC 2007 y 2012, cuyos pesos fueron descargados de [1] con una base de datos específica para coches autónomos[2] que se detallará en la siguiente sección. La segunda, el entrenamiento de la red desde un punto de partida pre-entrenado con el fin de mejorar el desempeño del sistema.

## 4. Resultados Experimentales

Esta sección se presentan los resultados de los experimentos llevados a cabo durante la realización de este trabajo, con el objetivo de validar el funcionamiento y desempeño del sistema desarrollado en base a las técnicas y métodos presentados en la Sec. 3. Se evalúa la precisión y velocidad del sistema sobre una de las bases de datos de [2] que se describirán más adelante en esta sección.

Se utilizan dos bases de datos específicamente recopiladas para desarrollar y evaluar sistemas de visión por computador para coches autónomos: *Udacity Self-Driving-Car Dataset 1* y *Udacity Self-Driving-Car Dataset 2*[2]. Éstas bases de datos fueron recolectadas por *Udacity*. Además de estas bases de datos se utilizan videos con similar contenido que las bases de datos para probar el sistema.

La primera base de datos *Udacity Self-Driving-Car Dataset 1* incluye imágenes obtenidas con una cámara colocada en la ventana frontal de un coche especialmente equipado para recopilación de otros datos importantes para los sistemas de conducción autónoma. Este set de datos fue recopilada al manejar durante el día bajo condiciones climáticas normales (sin lluvia) en las calles de la ciudad de Mountain View en California, Estados Unidos y en ciudades vecinas cercanas a ésta. El set contiene más de 65000 anotaciones realizadas sobre 9423 imágenes obtenidas a través de cámaras de la marca *Point Grey* (especialmente utilizadas para investigación) a máxima resolución de 1920x1200 a una frecuencia de 2Hz, es decir, capturando dos imágenes por segundo. Este set de datos fue anotado por el servicio ofrecido por la empresa *CrowdAI* que utiliza una combinación de anotaciones realizadas por personas y técnicas de aprendizaje de máquina. Ésta base de datos fue utilizada como *set de prueba*.

Este set contiene anotaciones de tres clases: (1) Coches, (2) Camiones, (3) Peatones.

A continuación, en la Fig. 4, se muestra un ejemplo de las imágenes que contiene la primera base de datos de *Udacity* y la Fig. 5 muestra los mismos ejemplos con anotaciones, éstas son los *ground truths* con los que se entrena a la red.

La segunda base de datos *Udacity Self-Driving-Car Dataset 2* es similar a la primera pero contiene anotaciones con un campo para identificar el nivel de oclusión de la imagen así como anotaciones adicionales referentes a las luces de tráfico. Los sets de datos fueron anotados solamente por personas utilizando *Autti*<sup>6</sup>. Esta base de datos está conformada por 15000 imágenes, sin embargo, solamente 13063 de ellas contienen anotaciones. Se descartan aquellas imágenes que no cuentan con anotación.

Este set contiene anotaciones de tres clases: (1) Coches, (2) Camiones, (3) Pea-

<sup>6</sup> <http://autti.co/>



Figura 4: Ejemplo de las imágenes de la base de datos *Udacity Self-Driving-Car Dataset 1*.



Figura 5: Ejemplo de las imágenes de la base de datos con anotaciones *Udacity Self-Driving-Car Dataset 1*.

tones, (4) Señales de tráfico, (5) Ciclista.

Se realizaron varios experimentos que abarcan las etapas desde el entrenamiento (dado que no existen trabajos publicados con la misma base de datos), validación y evaluación. Se utiliza como métrica la precisión que equivale a un valor porcentual que mide la precisión del sistema de detección. Esta precisión se obtiene para diferentes niveles de umbral de detección (*threshold*).

Para obtener esta métrica se programó un algoritmo en el lenguaje de programación Python. A grandes rasgos el procedimiento que se utilizó para el cálculo de la precisión frente al umbral de detección es el siguiente: (1) Se calcula la IOU entre las cajas delimitadoras obtenidas a través de la predicción del sistema entrenado y las anotaciones (*ground truths*) del set de prueba a utilizar. (2) Con cada una de las imágenes del set de prueba se realiza un conteo de todas las detecciones positivas identificando: verdaderos positivos y falsos positivos. (3) Se calcula la precisión del detector para diferentes niveles de umbral de detección. (4) Se calcula el recall con la siguiente fórmula para diferentes niveles de umbral de detección. (5) Se calcula el *f-score* del sistema.

Para todos los experimentos se utilizó un servidor con una tarjeta GPU Tesla K80 junto con los *drivers* pertinentes de nvidia. El sistema operativo del servidor es Ubuntu 14.04.5 LTS.

Para todos los experimentos se utilizaron los parámetros del sistema que se muestran en la Tab. 2 y los parámetros de entrenamiento se muestran en la Tab. 3. Estos valores son comunes para todos los experimentos.

Parámetro	Valor
Subdivisiones	8
Ancho	416
Alto	416

Tabla 2: Parámetros comunes en el entrenamiento del sistema.

Parámetro	Valor
Tamaño de <i>batch</i>	64
Coefficiente de aprendizaje	0.005, 0.0005*
Momentum	0.9
Decay	0.0005
Optimizador	RMSprop

Tabla 3: Parámetros utilizados en el entrenamiento del sistema. \*El coeficiente de aprendizaje se cambia a 0.0005 después de cierta cantidad de épocas. En cada experimento, esta cantidad varía.

Las subdivisiones hacen referencia a las divisiones verticales y horizontales que el sistema hace para generar un encuadre de  $S \times S$  de la imagen de entrada, donde  $S$  en este caso es 8. El alto y ancho hacen referencia al tamaño de entrada

de la imagen al sistema en píxeles.

Se empleó una arquitectura para todos los experimentos. La arquitectura definida para la red es la versión reducida de YOLOv2 que cuenta de 9 capas convolucionales, 6 capas de *max-pooling*, una capa de proposición de región (RPN) y en cada capa convolucional se aplica normalización del tipo *batch normalization* que ayuda en el entrenamiento y mejora la precisión.

Finalmente, para la evaluación de cada uno de los experimentos se utilizó el mismo set de datos. Las imágenes se obtuvieron de la base de datos *Udacity Self-Driving-Car Dataset 1*, de donde se extrajeron 3266 de ellas. Se escogió este número porque representa el 20 % de la base de datos *Udacity Self-Driving-Car Dataset 2*. La Fig. 6 se muestran la cantidad de objetos por clase presentes en el set de evaluación.

Al realizar la evaluación de las imágenes en el servidor, se logra obtener una velocidad de detección de entre 130fps y 145fps.

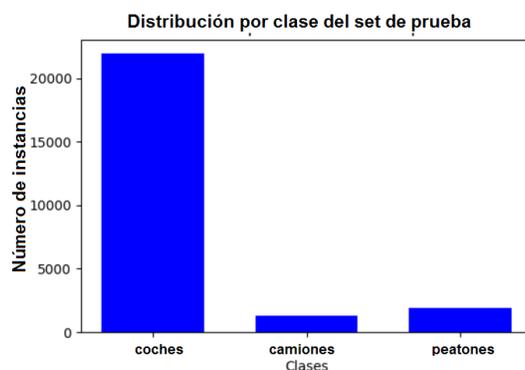


Figura 6: Distribución de aparición de objetos por clases en el set de evaluación.

#### 4.1. Experimento 1: 20 % Validación

Para este experimento se dividió la base de datos *Udacity Self-Driving-Car Dataset 2* en la siguiente proporción: 80 % para entrenamiento y 20 % para validación. Los resultados se presentan a continuación.

**Entrenamiento.** Para el entrenamiento de la red se utilizó inicialización aleatoria de pesos, en otras palabras, se entrenó al sistema *desde cero*. Los parámetros utilizados en el entrenamiento se presentan en la Tab. 3. Con estos parámetros se entrenó a la red por 10000 épocas, lo que tardó aproximadamente 4 días.

En la Fig. 7 se muestra la curva de pérdida de la red en el entrenamiento. La curva representa al promedio móvil de la pérdida por lo que es una curva suavizada. Se entrena al sistema hasta que éste no presenta más mejoras en cuanto a la pérdida donde se llega a un mínimo, ya sea local o global. La pérdida mínima promedio que se obtiene es de 8.25.

Se puede ver que la pérdida decrece a un ritmo aproximadamente constante hasta la época 4000 y luego la velocidad disminuye. En este punto, alrededor de 6000 épocas, es que se cambia el coeficiente de aprendizaje de 0.005 a 0.0005. Después de la época 10000 el sistema deja de mejorar asumiendo que se llega a un mínimo local (*settle point*) o global por lo que el entrenamiento se termina.

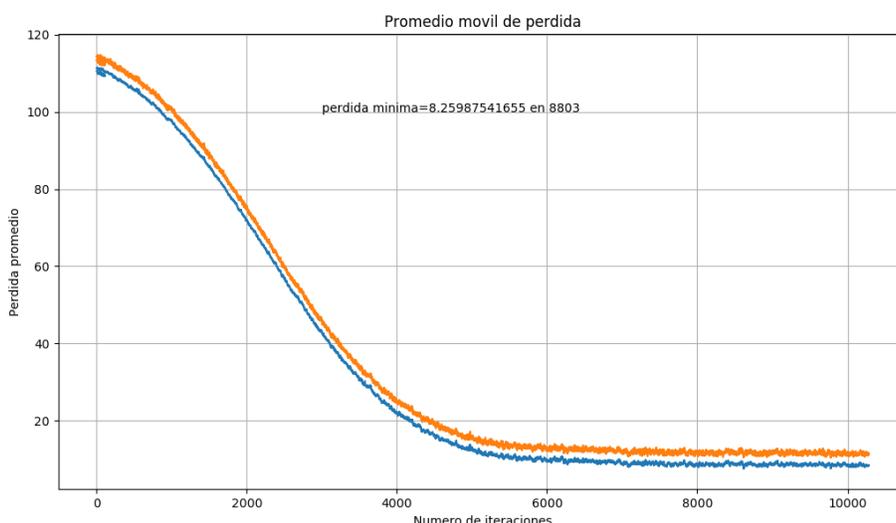


Figura 7: Resultados del entrenamiento del sistema en el experimento 1, pérdida promedio móvil del set de entrenamiento (azul) y pérdida móvil de los datos de validación (naranja)

Para poder realizar una comparación entre los experimentos realizados, se obtiene el  $f$ -score para 11 valores diferentes con separación constante de umbral de detección mostrado en la Fig. 8. La gráfica muestra estas métricas para cada valor de umbral. El valor máximo de  $f$ -score es 0.38 para un umbral de 0.1.

#### 4.2. Experimento 2: 10 % Validación

Para este experimento se dividió la base de datos *Udacity Self-Driving-Car Dataset 2* en la siguiente proporción: 90 % para entrenamiento y 10 % para validación. Los resultados se presentan a continuación.

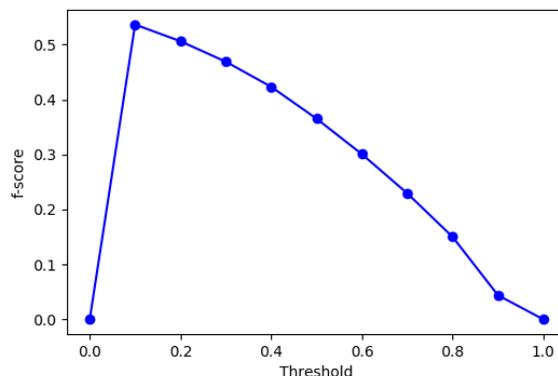


Figura 8:  $f$ -score en función de diferentes niveles de umbral de detección sobre el set de evaluación para el primer experimento.

**Entrenamiento.** Para el entrenamiento de la red se utilizó inicialización aleatoria de pesos. Con estos parámetros se entrenó a la red por 16000 épocas, lo que tardó aproximadamente 6 días.

En la Fig. 9 se muestra la curva de pérdida de la red en el entrenamiento. Se entrena al sistema hasta que éste no presenta más mejoras en cuanto a la pérdida donde se llega a un mínimo, ya sea local o global. La pérdida mínima promedio que se obtiene es de 6.72, menor que en el anterior experimento.

Se puede ver que la pérdida decrece a un ritmo aproximadamente constante hasta la época 6000 y luego la velocidad disminuye. En un punto futuro a éste (alrededor de 7000 épocas) es que se cambia el coeficiente de aprendizaje de 0.005 a 0.0005. Después de la época 16000 el sistema deja de mejorar por lo que el entrenamiento se termina.

**Análisis Cuantitativo** A continuación, se muestran la gráfica de  $f$ -score del sistema obtenido en este experimento utilizando el set de evaluación, Fig. 10. La gráfica muestra esta métrica para cada valor de umbral. El valor máximo de  $f$ -score es 0.52 para un umbral de 0.1.

#### 4.3. Experimento 3: Entrenamiento desde *Checkpoint*

Para este experimento se dividió la base de datos *Udacity Self-Driving-Car Dataset 2* en la siguiente proporción: 90% para entrenamiento y 10% para validación. Los resultados se presentan a continuación.

**Entrenamiento.** En este caso el entrenamiento fue diferente ya que se partió desde un modelo previamente entrenado utilizando un *checkpoint* o punto de

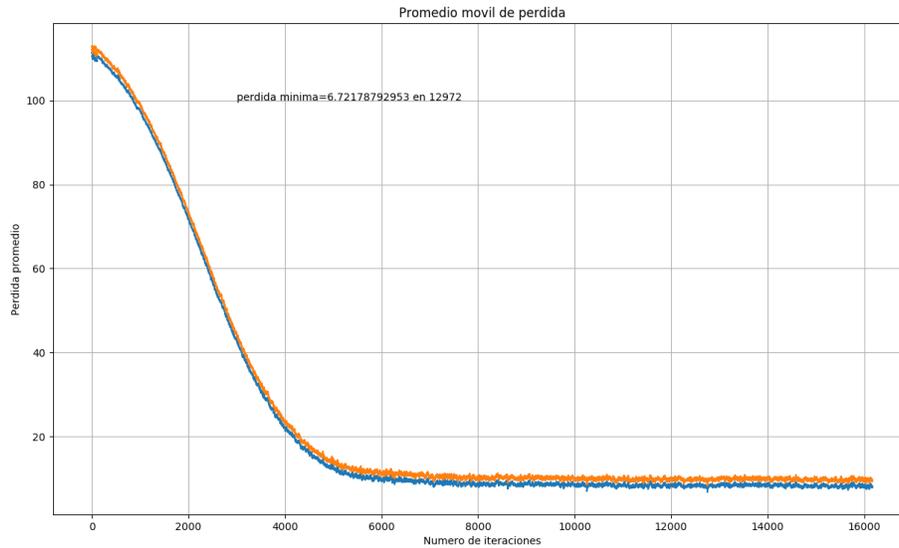


Figura9: Pérdida promedio móvil de la etapa de entrenamiento del sistema en el experimento 1

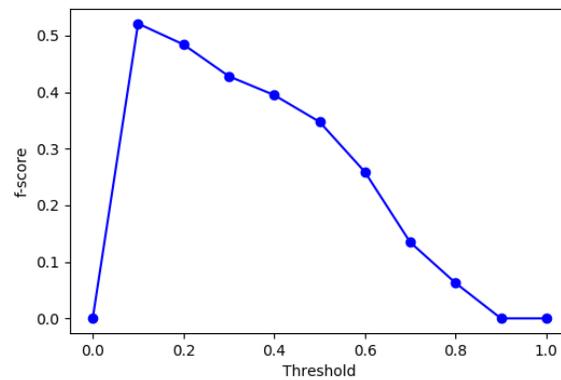


Figura10:  $f$ -score en función de diferentes niveles de umbral de detección sobre el set de evaluación para el segundo experimento.

partida<sup>7</sup> entrenado con las mismas bases de datos que se utilizan en este trabajo. El entrenamiento tardó aproximadamente 2 días, menos que en los casos anteriores ya que la pérdida inicial era del 10.25. Después del entrenamiento se

<sup>7</sup> Web de donde se obtuvo el *checkpoint*: <https://github.com/udacity/self-driving-car/tree/master/vehicle-detection/darkflow>

alcanzó una pérdida de 6.54. En este caso no se incluye la curva de entrenamiento ya que es similar a los casos anteriores y no aporta información nueva. Se entrenó a la red por 10237 épocas.

**Análisis Cuantitativo.** La Fig. 11 muestra la métrica de  $f$ -score para cada valor de umbral. El valor máximo de  $f$ -score es 0.53 para un umbral de 0.1.

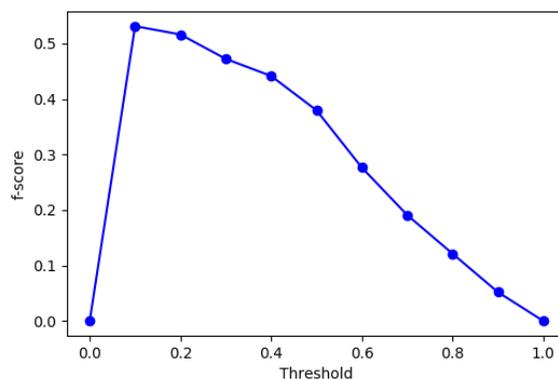


Figura 11:  $f$ -score en función de diferentes niveles de umbral de detección sobre el set de evaluación para el tercer experimento.

#### 4.4. Experimento 4: Fine-Tuning con un Modelo Pre-Entrenado

Para este experimento se dividió la base de datos *Udacity Self-Driving-Car Dataset 2* en la siguiente proporción: 90 % para entrenamiento y 10 % para validación. Los resultados se presentan a continuación.

**Entrenamiento.** En este caso el entrenamiento fue diferente ya que se partió desde un modelo previamente entrenado con la base de datos PASCAL VOC[3]. La arquitectura de red utilizada es la misma que en los casos anteriores. Sin embargo, al utilizar una inicialización de pesos con una red ya entrenada pero con otra base de datos, se pretende obtener un mejor punto de partida que en los experimentos de las Sec. 4.1 y 4.2. En este caso se utiliza la técnica de *transfer learning* que se discutió anteriormente en la Sec. 3.1. Los pesos utilizados para el entrenamiento fueron descargados de la web<sup>8</sup>. Luego se entrenó normalmente la red con nuestras bases de datos.

En este caso no se incluye la curva de entrenamiento ya que es similar a los casos anteriores y no aporta información nueva. Se entrenó a la red por 9125 épocas.

<sup>8</sup> Web de donde se obtuvieron los pesos: <https://pjreddie.com/darknet/yolo/>

**Análisis Cuantitativo.** La Fig. 12 muestra el  $f$ -score para cada valor de umbral. El valor máximo de  $f$ -score es 0.35 para un umbral de 0.1. La velocidad máxima alcanzada por el sistema al realizar las detecciones fue de 145fps.

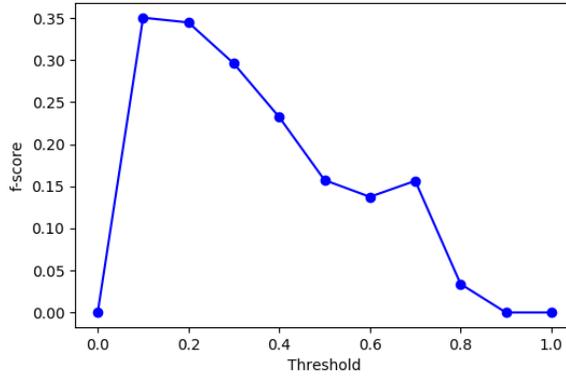


Figura 12:  $f$ -score en función de diferentes niveles de umbral de detección sobre el set de evaluación para el cuarto experimento.

En la Tab. 4 se presenta un breve resumen de los resultados obtenidos en el presente trabajo. La tabla presenta una comparación de resultados entre los sistemas obtenidos en los diferentes experimentos. Se incluyen el número del experimento seguido de una breve descripción del mismo junto con la proporción de set de datos de entrenamiento/validación. Se aprecia que los mejores resultados, tomando en cuenta un balance entre la precisión y el *recall*, es la red obtenida mediante el **experimento 3** con la cual se alcanza un puntaje de 0.53.

#### 4.5. Resumen de Resultados

En la Tab. 4 se presenta un breve resumen de los resultados obtenidos en el presente trabajo. La tabla muestra una comparación de resultados entre los sistemas obtenidos en los diferentes experimentos. Se incluyen el número del experimento seguido de una breve descripción del mismo junto con la proporción de set de datos de entrenamiento/validación. Se aprecia que los mejores resultados, según el  $f$ -score, es la red obtenida mediante el **experimento 3** con la cual se alcanza un puntaje de 0.53. Sin embargo, el resultado del **experimento 2** no se queda atrás alcanzando un puntaje de 0.52, comparable al experimento 3. El peor resultado se obtiene en el experimento 4 al utilizar *transfer learning* con un puntaje de 0.35. Finalmente, con un resultado similar al caso anterior, el experimento 1 presenta un puntaje de 0.38.

Experimento	f-score	máximo Umbral
1. Inicializacion aleatoria (80/20)	0.38	0.1
2. Inicializacion aleatoria (90/10)	0.52	0.1
3. Inicializacion <i>checkpoint</i> (90/10)	<b>0.53</b>	0.1
4. <i>Transfer learning</i> (90/10)	0.35	0.1

Tabla 4: Comparación de resultados entre los sistemas obtenidos en los diferentes experimentos. En la tabla se incluye el número del experimento seguido de una breve descripción del mismo junto con la proporción de set de datos de entrenamiento/validación.

## 5. Conclusiones

Las principales conclusiones que se derivan de los resultados teóricos y experimentales de este trabajo son las siguientes:

- Se ha propuesto utilizar una nueva base de datos para entrenar a un sistema de detección de objetos en tiempo real YOLO con el fin de estudiar su aplicación en el área de coches autónomos. Ningún trabajo publicado anteriormente estudia el entrenamiento y comportamiento del sistema con las bases de datos utilizadas en este trabajo.
- Se han evaluado diferentes técnicas de entrenamiento para la red y se ha realizado la comparación entre ellas. Se concluye que el mejor resultado se obtiene al utilizar un *checkpoint* alcanzando un mejor puntaje (*f-score*) de 0.53 con un umbral de detección del 10% como se demuestra **experimento 3**. Sin embargo, un resultado comparable se obtiene con el **experimento 2** llegando a un *f-score* de 0.52 mediante el entrenamiento de la red con inicialización de pesos aleatoria y utilizando la misma cantidad de datos para entrenamiento y validación que en el experimento 3.
- Se demuestra que al utilizar más datos de entrenamiento con inicialización aleatoria es posible mejorar los resultados del sistema propuesto. En el **experimento 2** se alcanza un puntaje considerablemente mayor que en el **experimento 1**, 0.52 frente a 0.38. Esto se debe a que la arquitectura utilizada, tiende a presentar *overfitting* con una cantidad de datos de entrenamiento del 80% de la base de datos, mientras que al utilizar 90% de datos se logra regularizar el entrenamiento y obtener un mejor resultado.
- Mediante un análisis cualitativo de las detecciones sobre las imágenes de las bases de datos utilizadas se observa que el sistema tiene dificultades al detectar objetos alejados de la cámara y aquellos con poca iluminación.
- Se ha podido observar que existen varios resultados en donde el sistema entrega más de una detección por objeto a pesar de que YOLOv2 integra la técnica de *non-max suppression* pensada para evitar ese tipo de resultados. Al analizar cuidadosamente los resultados, el entrenamiento y la base de datos de entrenamiento se atribuye este inconveniente a la base de datos. Se puede observar que en *Udacity Self-Driving-Car Dataset 2* las anotaciones

no son del todo correctas, existiendo en varios casos múltiples anotaciones por objeto con coordenadas levemente diferentes.

- Se ha comprobado el funcionamiento en tiempo real del sistema, el mismo demostró alcanzar velocidades de hasta 145 fps a la hora de llevar a cabo la detección en una GPU Tesla K80.

En base a los resultados experimentales y las conclusiones que de ellos se han extraído, las líneas de trabajo futuro que se proponen son las siguientes:

- Estudiar la aplicación de diferentes arquitecturas de red siguiendo el mismo sistema de detección de YOLO y estudiar el compromiso de la precisión frente a la velocidad. El presente estudio se limitó a estudiar una sola arquitectura de red dando prioridad a la velocidad de procesamiento. Posiblemente otras arquitecturas permitan una mayor precisión sin comprometer grandemente la velocidad.
- Estudiar los efectos de entrenar al sistema con otras bases de datos relacionadas con coches autónomos donde existan muchos objetos de las clases coche entre otros importantes para tal aplicación. Se pudo notar que no todas las anotaciones de las bases de datos [2] están hechas correctamente, por ejemplo, existen cajas delimitadoras repetidas varias veces para tan solo un objeto. Esto puede ser la causa de un bajo desempeño en el sistema. Por lo tanto, trabajar con otras bases de datos y estudiar el desempeño de la red en otras resultaría interesante.
- Diseño de un sistema que permita realizar el seguimiento de los objetos cercanos al coche a través de la secuencia de fotogramas. Por ejemplo, se puede inferir la velocidad del coche que pasa al lado y es detectado constantemente. Actualmente, se estudian y existen arquitecturas y sistemas de seguimiento que toman en cuenta no solamente el fotograma actual sino todos los anteriores añadiendo así al sistema de información temporal. Estos sistemas se basan, generalmente, en el uso de redes neuronales recurrentes (RNN) en especial la implementación de ellas, redes *long short term memory* (LSTM). Un ejemplo de un sistema de seguimiento basado en YOLO es ROLO[7].

## Referencias

1. "YOLO: Real-Time Object Detection." [Online]. Available: <https://pjreddie.com/darknet/yolo/>
2. "self-driving-car: The Udacity open source self-driving car project," Jul. 2017, original-date: 2016-10-02T02:47:23Z. [Online]. Available: <https://github.com/udacity/self-driving-car>
3. M. Everingham, L. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *Int. J. Comput. Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010. [Online]. Available: <http://dx.doi.org/10.1007/s11263-009-0275-4>
4. J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art," *CoRR*, vol. abs/1704.05519, 2017. [Online]. Available: <http://arxiv.org/abs/1704.05519>
5. T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: <http://arxiv.org/abs/1405.0312>
6. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015. [Online]. Available: <http://arxiv.org/abs/1512.02325>
7. G. Ning, Z. Zhang, C. Huang, Z. He, X. Ren, and H. Wang, "Spatially supervised recurrent convolutional neural networks for visual object tracking," *arXiv preprint arXiv:1607.05781*, 2016.
8. J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: <http://arxiv.org/abs/1506.02640>
9. J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016. [Online]. Available: <http://arxiv.org/abs/1612.08242>
10. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
11. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: <http://arxiv.org/abs/1409.4842>

## Autores



**Emilio S. Corchado Rodríguez** es profesor Titular acreditado como Catedrático de Universidad por la ANECA (Agencia Nacional de Evaluación de la Calidad y Acreditación) desde febrero de 2012. Ha sido elegido presidente del IEEE España durante el periodo 2014-2015. Es coautor de diversas patentes y propiedades intelectuales. Cuenta con una dilatada experiencia en la revisión y evaluación de proyectos de la Comisión Europea desde 2007.

Es coautor de más de 60 publicaciones en revistas científicas con alto índice de impacto y también actúa como miembro del Consejo de Redacción de varias revistas de reconocido prestigio internacional. Ha sido invitado como editor en varias “Special Issues” de alto renombre. Es el organizador de diversos congresos científicos como HAIS, IDEAL, SOCO, CISIS, NABIC, CASON, NWESP, WICT. Ha participado en varios proyectos Europeos (IT4Innovation, APP for AAL, ENGINE, SOFTCOMP, EMaCS, IntelliCIS, NISIS). Además ha sido Investigador Principal de diversos proyectos a nivel nacional y regional, también ha estado trabajando en estrecha colaboración con empresas como Grupo Antolín, Inmotia, Infranor, etc.

---

**Juan M. Corchado Rodríguez** es catedrático de Ciencia de la Computación e Inteligencia Artificial. Ha sido Vicerrector de Investigación y Transferencia de la Universidad de Salamanca y actualmente es Director del Grupo en Biotecnología, Sistemas Inteligentes y Tecnología Educativa (BISITE) de esta misma universidad (<http://bisite.usal.es>). Doctor en Informática por la Universidad de Salamanca, España (1998) y doctor en Inteligencia Artificial por la University of the West of Scotland, Reino Unido (2000). Es Coordinador Académico del Instituto de Arte y Tecnología de la Animación. Anteriormente ha sido Subdirector de la Escuela Superior de Ingeniería Informática en la Universidad de Vigo (España, 1999-00) e investigador en la Universidad de Paisley (1995-98). Colabora, como investigador, con el Laboratorio Oceanográfico de Plymouth (Reino Unido) desde 1993. Ha dirigido más de 90 proyectos de Inteligencia Artificial con financiación tanto internacional, como nacional o autonómica. Ha dirigido más de 20 tesis doctorales y es co-autor de más de 800 libros, capítulos de libros, artículos en revistas científicas, etc. la mayoría de ellos presentan aspectos tanto prácticos como teóricos relacionados con los Sistemas Híbridos de Inteligencia Artificial, la Biomedicina, la Inteligencia Ambiental, los Sistemas Inalámbricos y la Oceanografía. Es autor de más de una veintena de propiedades intelectuales y patentes.





**Jorge Durán Escudero**, natural de Salamanca, es Graduado en Ingeniería Informática por la Universidad de Salamanca. Su dedicación a la informática empezó con 12 años lo que le ha permitido ir evolucionando su carrera profesional desde escritor técnico a desarrollador de todo tipo de aplicaciones. Agraciado con varias becas tanto del Ministerio de Educación, como de su universidad, ha realizado estudios de maestría en Sistemas Inteligentes en 2017 en la Universidad de Salamanca. Sus principales trabajos se centran en las técnicas de Big Data y en la analítica visual como técnica para facilitar la comprensión de conjuntos de datos masivos.

---

**Lucía Fargallo Calvo**, nacida en Valladolid, completó la base de su educación en el año 2016 obteniendo el título de Ingeniera de Organización Industrial por la Universidad de Valladolid. En la Universidad de Salamanca amplió sus conocimientos con el Master Oficial en Sistema Inteligentes en el año 2017. Su trabajo se ha basado en la optimización de procesos de control utilizando algoritmos de la Inteligencia Artificial.



**Michael D. Galdámez Canales**, natural de Tegucigalpa, Honduras, completó su base de educación superior en el año 2015 obteniendo el título de Ingeniero en Ciencias de la Computación por la Universidad Católica de Honduras. En la Universidad de Salamanca amplió su formación realizando el Máster Oficial en Sistemas Inteligentes en 2017. Su trabajo se ha basado en la identificación biométrica basada en redes neuronales.

**Francisco José García Peñalvo** es profesor Titular de Universidad (acreditado a Catedrático de Universidad) del Departamento de Informática y Automática en la Universidad de Salamanca (USAL), con 3 sexenios de investigación y 4 quinquenios docentes reconocidos. Además, es Profesor Distinguido de la Escuela de Humanidades y Educación del Tecnológico de Monterrey, México. Desde 2006 es el director del Grupo de Investigación Reconocido por la USAL (y Grupo de Excelencia de Castilla y León) GRIAL (GRupo de investigación en InterAcción y eLearning). Es director de la Unidad de Investigación Consolidada de la Junta de Castilla y León (UIC 81). Ha sido Vicedecano de Innovación y Nuevas Tecnologías de la Facultad de Ciencias de la USAL entre 2004 y 2007 y Vicerrector de Innovación Tecnológica de esta Universidad entre 2007 y 2009. Actualmente es el Coordinador del Programa de Doctorado en Formación en la Sociedad del Conocimiento de la USAL. Para una información más detallada de las publicaciones estos son los enlaces públicos al perfil Scholar (<http://goo.gl/sDwrr0>) y ORCID (<http://orcid.org/0000-0001-9987-5584>).



**Angélica González Arrieta** es doctora en Informática por la Universidad de Salamanca. Cuenta con una amplia experiencia investigadora en el campo de sistemas conexionistas. Es Profesora Titular del Departamento de Informática y Automática de dicha Universidad, del área de Ciencia de la Computación e Inteligencia Artificial. Actualmente compatibiliza su labor docente e investigadora con la dirección de diversas actividades formativas sobre recursos tecnológicos, pericia y seguridad informática, colaborando activamente con las Fuerzas y Cuerpos de Seguridad.

**Diego Manuel Jiménez Bravo**, natural del municipio cántabro de Castro-Urdiales, cursó los estudios de Grado en Ingeniería Informática de Gestión y Sistemas de Información en la Universidad del País Vasco (UPV/EHU). Posteriormente obtuvo el título del Máster en Sistemas Inteligentes por la Universidad de Salamanca (USAL). En la actualidad cursa los estudios de doctorado en Ingeniería Informática en la USAL. Forma parte del Grupo de investigación BISITE, de la misma entidad, donde realiza labores de investigación. Sus principales intereses investigativos se centran en el campo de la minería de datos, machine learning, minería web, redes sociales, procesamiento de señales y optimización energética. Además, en dicho grupo, colabora en diversos proyectos de investigación centrados en sus líneas de investigación.





**Vivian Félix López Baptista** es profesora titular de la Universidad de Salamanca en el área de Ciencias de la Computación e Inteligencia Artificial. Doctorada en Informática por la Universidad de Valladolid en 1996. Miembro del Grupo de Minería de Datos. Ha realizado investigación en diferentes campos como procesamiento del lenguaje

natural, redes neuronales y minería de datos. Tiene 80 artículos publicados en revistas de reconocido prestigio, talleres y actas de conferencias, 20 libros y capítulos de libros y 20 informes técnicos, la mayoría de ellos en estos temas. Miembro del comité organizador y científico de varios simposios internacionales. Fue directora del Máster en Sistema Inteligente y del Programa de Doctorado en Informática y Automática de la Universidad de Salamanca desde junio de 2010 hasta octubre de 2012.

---

**Lucía Martín Gómez**, natural de Salamanca, finalizó los estudios superiores en 2016 obteniendo el título de Graduada en Ingeniería Informática por la Universidad de Salamanca. En la misma universidad amplió su formación realizando el Máster de Sistemas Inteligentes durante el curso 2016-2017. Adicionalmente, en el año 2012, obtuvo el título de violinista en el Conservatorio Profesional de Salamanca. Actualmente cursa sus estudios de doctorado en Ingeniería Informática conjugando sus conocimientos musicales con la Inteligencia Artificial. Forma parte del Grupo de Investigación BISITE del Departamento de Informática y Automática de la Universidad de Salamanca. En el ámbito de la investigación ha participado en diversos congresos internacionales y colaborado en algunos proyectos de investigación relacionados con el Internet de las Cosas y la Industria 4.0 entre otros.



**Yeray Mezquita Martín** es graduado en Ingeniería Informática por la universidad de Salamanca en el año 2015. Máster en Sistemas Inteligentes también por la Universidad de Salamanca, en el año 2017. Actualmente, estudiante de doctorado de primer año. Desde hace tres años trabajando como investigador y desarrollador en el grupo de investigación de la universidad de Salamanca BISITE.

Entre los trabajos realizados, se encuentra la selección de características y clasificación de datos bioinformáticos. Experiencia en el ámbito del control, optimización y simulación de procesos, así como en el de desarrollo de prototipos IoT para la optimización de los recursos en el ámbito agroganadero. De forma más actual, y conforme al tema del doctorado, se encuentra realizando trabajos de investigación y desarrollo en la tecnología blockchain.

**María N. Moreno García** es Catedrática de Universidad del Departamento de Informática y Automática de la Universidad de Salamanca, directora del grupo de investigación en Minería de Datos y Coordinadora del Programa de Doctorado de Ingeniería Informática. Sus áreas de investigación de interés se centran en el desarrollo y aplicación de algoritmos de minería de datos en diferentes dominios como minería web y de medios sociales o apoyo a las decisiones en medicina.



**María Navarro Cáceres**, Doctorada en ingeniería informática por la universidad de salamanca, comenzó a trabajar en dicha universidad en 2012 bajo una beca de colaboración. Posteriormente, pudo cursar su doctorado a través de un contrato predoctoral dentro del programa de formación de personal universitario (FPU) promovido por el ministerio de educación y cultura (MECD). Actualmente, está vinculada a la universidad de Salamanca mediante un

contrato de profesor asociado. Muchas de las actividades realizadas en todos los ámbitos antes y durante la formación predoctoral, han supuesto la constitución de una línea de investigación coherente y bien definida, que abarca los ámbitos de la informática y la música. En el campo de la investigación, actualmente, se encuentra desarrollando una investigación centrada en la composición musical automática y en el análisis de características musicales a partir de la inteligencia artificial. Ha participado en proyectos de tecnología educativa, y de construcción musical (Proyecto MUDEI), entre otros. La mayoría de las publicaciones destacadas ponen en relieve la transversalidad de la música desde un punto de vista informático.

**Belén Pérez Lancho** es Licenciada y Doctora en Ciencias Físicas por la Universidad de Salamanca. Realizó su formación posdoctoral en la Universidad Paris VI (curso 1996-97) y desde 1998 es Profesora Titular en el Área de Ingeniería de Sistemas y Automática de la Universidad de Salamanca. Imparte o ha impartido docencia en las titulaciones de Ingeniería Informática, Física, Estadística y en el Máster en Sistemas Inteligentes. Pertenece al grupo de investigación BISITE y ha participado en más de 20 proyectos de investigación, principalmente en aplicaciones de sistemas de control y sistemas multiagente, colaborando en varias publicaciones científicas, en la dirección de tres tesis y de numerosos proyectos de fin de carrera. Ha ocupado cargos de gestión en la Facultad de Ciencias como Secretaria Académica (años 2000 a 2004) y como Vicedecana (años 2008 a 2016).





**Juan Francisco de Paz Santana**, Doctor en Informática y Automática por la Universidad de Salamanca en 2010. Ingeniero técnico en Informática y de Sistemas (2003), Ingeniero Informático (2005) y Diplomado en Estadística (2007) por la Universidad de Salamanca. Actualmente es profesor titular en Departamento de Informática y Au-

tomática de la Universidad de Salamanca. Además ha realizado el Máster Universitario en el Desarrollo de Sistemas para el Comercio Electrónico y el Máster Universitario en Animación Digital. Ha obtenido diferentes premios a nivel académico como el Premio de Grado, Premio de doctorado y en otros ámbitos como un premio de SUN Microsystems y otro de la Junta de Castilla y León. En la actualidad trabaja en el campo de los sistemas distribuidos basados en sistemas multiagentes, la Inteligencia Artificial, la bioinformática, la inteligencia ambiental etc. Ha participado en 48 proyectos de investigación a nivel regional, nacional e internacional. Es coautor de más de 180 artículos publicados en diversas revistas de ámbito nacional e internacional, workshops y simposios, 50 de estas publicaciones en revistas con índice de impacto. Ha sido organizador de numerosas conferencias y además ha sido revisor en diferentes conferencias y revistas de ámbito internacional en el área de las Ciencias de la Computación e Inteligencia Artificial. Es miembro del Grupo de Investigación BISITE, miembro del grupo IBASAL y socio de la AEPIA. Evaluador de la ANECA desde 2010 y FONCyT en 2011.

---

**Javier Pérez Marcos**, natural de Salamanca. Finalizó sus estudios superiores en 2016 obteniendo el título de Graduado en Ingeniería Informática por la Universidad de Salamanca. Además, amplió su formación de postgrado en dicha universidad, obteniendo el título de Master en Sistemas Inteligentes en el año 2017 y continuando sus estudios de segundo ciclo en el Doctorado en Ingeniería Informática. Trabaja como investigador en el grupo de investigación BISITE de la Universidad de Salamanca y colabora con el grupo de investigación MIDA de la misma universidad. Sus trabajos se centran en la minería de datos, más en concreto en los sistemas de recomendación.




---

**Silvana Roxani Revollar Chávez**, Profesora Ayudante Doctor en el área de Ingeniería de Sistemas y Automática: Automatización y Control.

**William Raveane** obtuvo una Licenciatura en Física por Southern Oregon University en Estados Unidos, y un Máster en Sistemas Inteligentes por la Universidad de Salamanca en España. Trabaja como programador principal en una compañía en Tokio, Japón, desarrollando sistemas cloud para aplicaciones web y procesamiento de imágenes. Sus principales líneas de investigación son las Redes Neuronales Profundas, en especial su uso en la visión artificial.



**Roberto Theron Sánchez** cursó sus estudios de Informática en la Universidad de Salamanca (Diplomatura) y la Universidad de la Coruña (Licenciatura). Tras entrar a formar parte del Grupo de Investigación en Robótica de la Universidad de Salamanca, presentó su trabajo de Tesis recibiendo el Premio Extraordinario de Doctorado. Posteriormente ha obtenido los títulos de Licenciado en Comunicación Audiovisual (Universidad de Salamanca) y Licenciado en Humanidades (Universidad de Salamanca).

En la misma Universidad de Salamanca continúa realizando su trabajo de investigador, como encargado del grupo VisUsal (dentro del Grupo de Investigación Reconocido GRIAL) que se centra en la combinación de enfoques procedentes de la Informática, Estadística, Diseño Gráfico y Visualización de Información, para obtener una adecuada comprensión de conjuntos de datos complejos. En los últimos años, se ha dedicado al desarrollo de herramientas de visualización avanzada para datos multidimensionales. En el área de Analítica Visual desarrolla productivas colaboraciones con grupos e instituciones de reconocido prestigio internacional, como el Laboratorio de Ciencias del Clima y del Medio Ambiente (París) o la Academia de Ciencias de Austria. Es autor de más de 100 artículos en revistas y congresos internacionales.

**Nicolás Ugrinovic Kehdi** nacido en hermosa ciudad de La Paz en Bolivia se graduó con honores en Ingeniería Electrónica por la Escuela Militar de Ingeniería. Gracias a una beca de excelencia académica otorgada por la fundación Simón I. Patiño con base en Ginebra (Suiza) es que tuvo la oportunidad de cursar el Máster en Sistemas Inteligentes en la Universidad de Salamanca. Posteriormente, realizó una estancia de investigación en el Instituto de Robótica e Informática de la Universidad Politécnica de Catalunya. Su trabajo se ha centrado en la aplicación de técnicas novedosas de Deep Learning en el área de Visión por Computador. Actualmente, es coordinador de las carreras de Mecatrónica y Sistemas Electrónicos de la Universidad Católica Boliviana en la ciudad de Santa Cruz.





**Pastora Isabel Vega Cruz** es Catedrática de Universidad del área de Ingeniería de Sistemas y Automática y lidera el Grupo de Investigación Reconocido de la USAL denominado “Supervisión y Control de Procesos”. La actividad del mismo se centra en proponer soluciones eficientes para la optimización y el control de procesos industriales,

integradas en los niveles más altos de la jerarquía del control y con el objeto de incrementar la seguridad, la fiabilidad y la calidad en la operación y en el control en las industrias de procesos. Concretamente, se investigan soluciones teóricas y prácticas, para la optimización global de la operación, el control avanzado de los procesos y la supervisión inteligente de los mismos. Entre las líneas de investigación más destacadas se encuentran: Control Predictivo Distribuido basado en sistemas multiagente, Optimización, supervisión y control de EDARs y Síntesis y diseño integrado de procesos. Las aplicaciones industriales más importantes incluyen los Sistemas Integrados de Aguas y los Sistemas de Poligeneración.

**Gabriel Villarrubia González** es Profesor Ayudante Doctor de la Universidad de Salamanca en el Departamento de Informática y Automática. Ingeniero Informático por la Universidad Pontificia de Salamanca (2011), habiendo estudiado el Máster en Sistemas Inteligentes en la Universidad de Salamanca (2012). Además ha realizado el Máster Universitario en Seguridad e Internet (2014) y el Máster en Dirección de Sistemas de Información (2015). A lo largo



de su formación ha seguido una línea de investigación bien definida, centrada en la aplicación de los sistemas multi-agente a los entornos de inteligencia ambiental, con especial atención a la definición de arquitecturas inteligentes y a la fusión de información. Ha participado en más de 30 proyectos de investigación, colaborando en la obtención de 15 propiedades intelectuales, habiendo realizado estancias internacionales. En el ámbito científico, es autor de más de 40 publicaciones científicas y codirector de 50 proyectos de fin de grado. Es miembro del comité científico de diferentes congresos, siendo miembro del comité organizador de (ISAMI, PACBB, MIS4TEL, PAAMS, DCAI). A nivel profesional fue nombrado Microsoft Student Partner en el año 2010. Ha ganado diferentes premios en el desarrollo de aplicativos móviles. Actualmente centra sus esfuerzos en diseñar una arquitectura multi-agente especialmente diseñada para la fusión dinámica de información en entornos inteligentes caracterizados por estar dotados de sensores heterogéneos que capturan información del entorno.