

# Progresos actuales en la Ingeniería del Lenguaje y del Conocimiento

Mireya Tovar Vidal  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
María Josefa Somodevilla García

Editores



# Progresos actuales en la Ingeniería del Lenguaje y del Conocimiento

# Progresos actuales en la Ingeniería del Lenguaje y del Conocimiento

Mireya Tovar Vidal  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
María Josefa Somodevilla García  
**Coordinadores**



Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación  
2020

Primera Edición **2020**  
ISBN digital: 978-607-8728-28-2  
ISBN impreso: 978-607-8728-25-1

DR © Benemérita Universidad Autónoma de Puebla  
4 Sur 104, Col. Centro Histórico, Puebla, Pue. CP 72000  
Teléfono: 01 (222) 229 55 00  
[www.buap.mx](http://www.buap.mx)

Montiel & Soriano Editores S.A. de C.V.  
15 sur 1103-6, Col. Centro  
Puebla, Pue.

Facultad de Ciencias de la Computación  
14 sur esq. Con Av. San Claudio  
Ciudad Universitaria, Puebla, Pue.  
Telfonos: 01 (222) 229 55 00 Ext. 7200 y 7204  
[www.cs.buap.mx](http://www.cs.buap.mx)

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA • *Rector:* José Alfonso Esparza Ortiz • *Secretario General:* Guadalupe Grajales Y Porras • *Vicerrector de Extensión y Difusión de la Cultura:* José Carlos Bernal Suárez • *Director General de Publicaciones:* Hugo Vargas Comsille • *Directora de la Facultad de Ciencias de la Computación:* María del Consuelo Molina García

Hecho en México  
*Made in Mexico*



## Prólogo

Este libro presenta diferentes tópicos de investigación en el área de la ingeniería del Lenguaje y del Conocimiento en las Ciencias de la Computación. La obra incluye veintiún capítulos de investigación en las áreas de reconocimiento de patrones en grafos, ontologías, visión computacional, aprendizaje automático, procesamiento de lenguaje natural, entre otras.

Los capítulos que forman parte de esta obra fueron escritos por investigadores y colaboradores de diferentes instituciones del país. Cabe mencionar que cada capítulo ha sido revisado por expertos en el área.

Finalmente, expresamos nuestro agradecimiento a los autores de cada capítulo por su valiosa aportación, a nuestros revisores por su invaluable labor, a la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla y a todos aquellos cuya participación contribuyó a la publicación de este libro.

Los editores,  
Mireya Tovar Vidal  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
María Josefa Somodevilla García



# Índice general

<b>Prólogo</b> .....	IV
<b>Capítulo 1.</b> Reconociendo patrones básicos 3-coloreables en grafos planares .....	1
<i>Cristina López, Guillermo De Ita</i>	
<b>Capítulo 2.</b> Diseño de recursos para la evaluación de una ontología de dominio pedagógico .....	12
<i>Yuridiana Alemán, María Somodevilla, Darnes Vilarriño</i>	
<b>Capítulo 3.</b> Etiquetado de video e identificación de unidades de acción para el reconocimiento de emociones centradas en el aprendizaje .....	20
<i>Yesenia Nohemí González-Meneses, Josefina Guerrero-García</i>	
<b>Capítulo 4.</b> Avances en la detección y predicción de trayectorias de obstáculos en ambientes vehiculares mediante visión computacional .....	30
<i>Lauro Reyes-Cocolezzi, Ivan Olmos-Pineda, Arturo Olvera-López</i>	
<b>Capítulo 5.</b> Grafos embebidos y el cálculo eficiente de su Número de Conjuntos Independientes .....	40
<i>Miguel Rodríguez-Hernández, Guillermo De Ita-Luna</i>	
<b>Capítulo 6.</b> Modelo de Datos de Consumo Eléctrico para una Función de Optimización .....	50
<i>Pérez Camacho Blanca Nydia, González Calleros Juan Manuel</i>	
<b>Capítulo 7.</b> Desarrollo de la competencia lectora con apoyo de las TIC .	61
<i>Adelina Escobar Acevedo, Josefina Guerrero García</i>	
<b>Capítulo 8.</b> Actualización de la Interfaz de Control y Monitoreo del Piano Cósmico .....	69
<i>Tonatiuh García Chávez, Arturo Fernández Téllez, Guillermo Tejada Muñoz</i>	
<b>Capítulo 9.</b> Extracción de Características de la Postura Corporal como una Variable Correlacionada con el Nivel de Atención .....	76
<i>Alfredo Garcia, Juan Manuel González y Amparo Palomino</i>	
<b>Capítulo 10.</b> Protocolo para Realización de Experimentos de Elicitación de Gestos Relacionados a Tareas Abstractas .....	86
<i>David Céspedes-Hernández, Juan Manuel González</i>	

<b>Capítulo 11.</b> Estudio de la cinemática de un robot de 18 grados de libertad.....	97
<i>Miguel Angel Ortega Palacios, Amparo Dora Palomino Merino, José Fernando Reyes Cortés</i>	
<b>Capítulo 12.</b> Búsqueda informada de coeficientes de la ecuación de movimiento de un quadricópteros mediante redes neuronales convolutivas	107
<i>Carlos Leopoldo Carreón Díaz de León, Sergio Vergara Limon, M. Aurora D. Vargas Treviño</i>	
<b>Capítulo 13.</b> Implementación de redes neuronales para el cálculo automático de los parámetros de configuración de un robot cartesiano ...	117
<i>Daniel Marcelo González Arriaga, María Aurora Diozcora Vargas Treviño, Sergio Vergara Limon</i>	
<b>Capítulo 14.</b> Caso: Identificación de Transacciones de Criptomonedas de Delitos Cibernéticos en Tor .....	125
<i>Julio Jesús Salas Conde, Manuel Martín Ortíz</i>	
<b>Capítulo 15.</b> Análisis sintáctico de H RG's .....	135
<i>Yolanda Moyao, Darnes Vilariño, Carlos Guillén and José de Jesús Lavalle</i>	
<b>Capítulo 16.</b> Preprocesamiento de imágenes para la autenticación de personas .....	145
<i>Aida A. Aparicio-Arroyo, Ivan Olmos-Pineda, J. Arturo Olvera-López</i>	
<b>Capítulo 17.</b> Análisis de técnicas para el Reconocimiento Automático de Lengua de Señas .....	155
<i>Daniel Sánchez-Ruiz, J. Arturo Olvera-López, Ivan Olmos-Pineda, Soraia Prietch</i>	
<b>Capítulo 18.</b> Minería y análisis de sentimientos de textos históricos: una revisión de la literatura .....	165
<i>Stephanie Vázquez-González, María Somodevilla-García</i>	
<b>Capítulo 19.</b> Evaluación comparativa de sistemas de reconocimiento automático del habla .....	177
<i>Víctor Manuel Morales-de-Jesús, María Josefa Somodevilla-García</i>	
<b>Capítulo 20.</b> Optimización para mejora de imágenes médicas .....	185
<i>Eustolia Carreón-Esteban, J. Arturo Olvera-López, Ivan Olmos-Pineda</i>	
<b>Capítulo 21.</b> Transformada de Fourier en tiempo corto para el análisis de espectrogramas .....	195
<i>Ricardo Ramos Aguilar, José Arturo Olvera López e Ivan Olmos Pineda</i>	

<b>Índice de Autores</b> .....	205
<b>Compiladores</b> .....	206
<b>Revisores</b> .....	207
<b>Editores</b> .....	208

# Capítulo 1

## Reconociendo patrones básicos 3-coloreables en grafos planares

Cristina López, Guillermo De Ita  
Facultad Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla  
Puebla, México  
cristyna2001@hotmail.com, deita@cs.buap.mx

**Resumen.** Este artículo se enfoca en el reconocimiento de un patrón básico que conforman a los grafos planares, que son los grafos rueda con diferentes tipos de centros y/o caras de eje. Nos centramos en el análisis del coloreo de vértices sobre estos patrones gráficos, e identificamos casos para la 3 o 4 colorabilidad de las ruedas. También consideramos diferentes composiciones entre ruedas y analizamos su proceso de colorabilidad.

Mostramos que para ciertas composiciones de ruedas  $G$ , existe un 3-coloreo válido que puede capturarse usando restricciones sobre variables simbólicas. Estas restricciones se expresan mediante una forma normal conjuntiva  $F_G$ . Demostramos que la satisfactibilidad de  $F_G$  implica la existencia de un 3-coloreo válido para  $G$ . De lo contrario, es necesario utilizar 4 colores para colorear correctamente  $G$ . La revisión de la Satisfactibilidad de  $F_G$  se puede realizar en tiempo polinomial aplicando resolución unitaria y propiedades generales sobre igualdad y desigualdad entre variables simbólicas..

**Keywords:** Problema de satisfactibilidad, 3-coloreo, teorema de los 4 colores, grafos de ruedas, grafo planar.

### 1 Introducción

El coloreo de grafos consiste en colorear los vértices de un grafo con el menor número de colores posibles, de forma que dos vértices adyacentes no pueden tener un mismo color. Si existe tal coloreo con  $k$  colores, se dice que el grafo es  $k$ -coloreable. El número cromático de un grafo  $G$ , denotado como  $X(G)$ , representa el número mínimo de colores para un coloreo válido de  $G$ . El problema del  $k$ -coloreo consiste en determinar si un grafo de entrada es  $k$ -coloreable.

Se han realizado varios trabajos para determinar cuándo un grafo es 3-coloreable, porque se ha demostrado que el problema de los 3 colores es un problema difícil (un problema NP-completo). Los resultados más recientes para determinar la 3-colorabilidad de un grafo se

basan en reconocer si el grafo de entrada es planar y sin triángulos (Grötzsch, 1959, p. 109), o descubriendo alguna relación entre la topología del grafo y sus partes estructurales (Stacho, 2010, p. 144; Lozin y Malyshev, 2017, p. 273; Golubic, 2004; Dvorač, Král y Thomas, 2009, p. 120; Fomin y Kratsch, 2010; Mertzios, y Spirakis, 2013, p. 332).

El coloreo de vértices de grafos es un campo activo de investigación con muchos subproblemas interesantes. El problema de coloreo de grafos tiene muchas aplicaciones en áreas como problemas de programación, asignación de frecuencia, planificación, etc. (Dvorač et al, 2009; Mertzios y Spirakis, 2013, p. 332). Los grafos planares juegan un papel importante tanto en la teoría de los grafos como en el área de dibujo de grafos. De hecho, los grafos planares tienen varias propiedades interesantes: son dispersos, cuatro coloreables, y su estructura interna se describe de manera sucinta y elegante (Cortese y Patrignani, 2004, p. 1).

La complejidad computacional inherente asociada con la resolución de problemas NP-difíciles ha motivado la búsqueda de métodos alternativos, que permiten que algunas instancias de problemas NP difíciles se resuelvan en tiempo polinomial. Por ejemplo, en el caso del problema de coloreo de vértices, el 2-coloreo se puede resolver en tiempo polinomial. Además, para determinar la 3-colorabilidad para grafos AT-free y para grafos perfectos, así como para determinar  $\chi(G)$  para algunas clases de grafos, tales como: grafos intervalo, grafos chordal y grafos de comparabilidad (Stacho, 2010, p. 144). En todos esos casos, se han encontrado estructuras especiales (patrones) para caracterizar las clases de grafos que son coloreables en complejidad de tiempo polinomial.

Proponemos un método novedoso para el reconocimiento de patrones de 3 o 4 colores en grafos planares. Esta propuesta se basa en la especificación lógica de las restricciones dada por un 3-coloreo válido en el grafo de patrón básico (ruedas) formando un grafo planar.

## 2 Preliminares

Sea  $G = (V, E)$  un grafo simple no dirigido (es decir, finito, sin bucles y sin múltiples aristas) con el conjunto de vértices  $V$  (o  $V(G)$ ) y el conjunto de aristas  $E$  (o  $E(G)$ ). Dos vértices  $v$  y  $w$  son llamados adyacentes, si  $\{v, w\} \in E$ , y la arista  $\{v, w\}$  es incidental a  $u$  y  $v$ . Dos aristas son adyacentes si tienen un vértice en común.

La vecindad de  $x \in V$  es  $N(x) = \{y \in V: \{x, y\} \in E\}$  y su vecindad cerrada es  $N(x) \cup \{x\}$ , que se denota por  $N[x]$ . Denotamos la cardinalidad de un conjunto  $A$  por  $|A|$ . El grado de un vértice  $x \in V$ , denotado por  $\delta(x)$ , es  $|N(x)|$ . El grado máximo de  $G$ , o simplemente el grado de  $G$ , es  $\Delta(G) = \max\{\delta(x): x \in V\}$ .

Un camino desde un vértice  $v$  a  $w$  es una secuencia de aristas:  $v_0v_1, v_1v_2, \dots, v_{n-1}v_n$  tal que  $v = v_0, v_n = w, v_k$  es adyacente a  $v_{k+1}$  y la longitud del camino es  $n$ . Un camino simple es un camino tal que  $v_0, v_1, \dots, v_{n-1}, v_n$  son todos diferentes. Un ciclo es simplemente un camino no vacío en el que el primer y el último vértice son idénticos; y un ciclo simple es un ciclo en el que no se repite vértice, excepto el primero y el último vértice. Un  $k$ -ciclo es un ciclo

de longitud  $k$  (tiene  $k$  aristas). Un ciclo de longitud impar se denomina ciclo impar, mientras que un ciclo de longitud par se denomina ciclo par. Un grafo sin ciclos se llama acíclico.

Dado un subconjunto de vértices  $S \subseteq V$ , el subgrafo de  $G$  donde  $S$  es el conjunto de vértices y el conjunto de aristas es  $\{u, v\} \in E: u, v \in S\}$ , se llama el subgrafo de  $G$  inducido por  $S$  y se denota por  $G|S$ .  $G - S$  denota el grafo  $G|(V - S)$ . El subgrafo inducido por  $N(v)$  se denota como  $H(v) = G|N(v)$ , el cual contiene todos los nodos de  $N(v)$  y todas las aristas que los conectan.

Un conjunto independiente o estable es un conjunto de vértices en un grafo donde ninguno de sus vértices es adyacente a otro. Es decir, es un conjunto  $S \subseteq V(G)$  de vértices tal que para cualquier par de ellos no hay una arista que los conecte. El tamaño de un conjunto independiente es el número de vértices que contiene. Un conjunto independiente  $S$  es maximal si no es un subconjunto adecuado de otro conjunto independiente, y es máximo si no hay otro conjunto independiente en  $G$  con una cardinalidad mayor que  $|S|$ .

El coloreo de un grafo  $G = (V, E)$  es una asignación de colores a sus vértices. El coloreo es válido si los vértices adyacentes siempre tienen diferentes colores. Un  $k$ -coloreo de  $G$  es un mapeo de  $V$  en el conjunto  $\{1, 2, \dots, k\}$  de  $k$  "colores". El número cromático de  $G$  denotado por  $\chi(G)$  es el valor mínimo  $k$  tal que  $G$  tiene un  $k$ -coloreo adecuado. Si  $\chi(G) = k$ , entonces se dice que  $G$  es  $k$ -cromático o  $k$ -coloreable.

Sea  $G = (V, E)$  un grafo.  $G$  es un grafo bipartito si  $V$  se puede dividir en dos subconjuntos  $U_1$  y  $U_2$ , llamados conjuntos parciales, de modo que cada arista de  $G$  se une a un vértice de  $U_1$  a un vértice de  $U_2$ . Si  $G = (V, E)$  es un grafo  $k$ -cromático, entonces es posible dividir  $V$  en  $k$  conjuntos independientes  $V_1, V_2, \dots, V_k$ , llamadas clases de color, pero no es posible dividir  $V$  en  $k - 1$  conjuntos independientes.

### 3 Grafos Planares

Un dibujo  $\Gamma$  de un grafo  $G$  asigna cada vértice  $v$  a un punto distinto  $\Gamma(v)$  del plano y cada arista  $\{u, v\}$  a una apertura simple de curva de Jordan  $\Gamma(u, v)$  con los puntos finales  $\Gamma(u)$  y  $\Gamma(v)$ . Un dibujo es planar si no se cruzan dos aristas distintas, excepto, posiblemente, en puntos finales comunes. Un grafo es planar si admite un dibujo planar. Un dibujo planar divide el plano en regiones conectadas llamadas caras. La cara sin límites usualmente es llamada *cara externa* o cara del exterior. Si todos los vértices son incidentales a la cara exterior, el grafo es llamado *outerplanar*.

Un grafo no conectado es planar si y solo si todos sus componentes conectados son planares. Quizás la propiedad más reconocida es la que establece el Teorema de Euler, que muestra que los grafos planares son escasos. Es decir, dado un grafo plano con  $n$  vértices,  $m$  aristas y  $f$  caras, tenemos  $n - m + f = 2$ . Un corolario simple es que para un grafo planar maximal con al menos tres vértices, donde cada cara es un triángulo ( $2m = 3f$ ), tenemos  $m = 3n - 6$ , y, por lo tanto, para cualquier grafo planar tenemos  $m \leq 3n - 6$ . Este número se



reduce a  $m = 2n - 3$  para grafos *outerplanar* máxima con al menos tres vértices (y  $m \leq 2n - 3$  para grafos *outerplanar* generales). Además, si  $n \geq 3$  y el grafo no tiene un ciclo de longitud 3, entonces  $m \leq 2n - 4$ . Finalmente, si el grafo es un árbol, entonces  $m = n - 1$ .

La primera caracterización de grafos planares se debe a Kuratowski (1930) “quien afirma que un grafo es planar si y solo si no contiene un subgrafo que es una subdivisión de  $K_5$  o  $K_{3,3}$ , donde  $K_5$  es el grafo completo de orden 5 y  $K_{3,3}$  es el grafo completo bipartito con 3 vértices en cada uno de los conjuntos de la partición” (p. 271). Un resultado similar, reescrito en términos de grafo minors, es el teorema de Wagner’s (1937) “que afirma que un grafo  $G$  es planar si y solo si no tiene  $K_5$  o  $K_{3,3}$  como minors. Esto significa que  $K_5$  o  $K_{3,3}$  no se puede obtener de  $G$  al contraer algunas aristas, eliminando algunas aristas, ni eliminando algunos vértices aislados” (p. 570). Obsérvese que las dos caracterizaciones son diferentes ya que un grafo puede admitir  $K_5$  como minor sin tener un subgrafo que es una subdivisión de  $K_5$ .

De ahora en adelante, consideramos solamente grafos planares como grafos de trabajo. Cada grafo planar sin triángulo es 3-coloreable (Grötzsch, 1959, p. 109) y se puede encontrar el 3-coloreo en tiempo lineal (Dvorák et al, 2009, p. 120). El teorema de los 4 colores (4CT) afirma que cualquier grafo planar es 4 coloreable (Robertson, Sanders, Seymour y Thomas, 1997, p. 2). Esto resuelve una conjetura que fue durante más de un siglo el problema más famoso no resuelto en la teoría de grafos y quizás en todas las matemáticas (Harary, 1969). Además 4CT proporciona un algoritmo en tiempo  $O(n^2)$  para 4 colorear cualquier grafo planar (Robertson et al 1997). Sin embargo la prueba actual conocida para el 4CT es asistida por computadora. Además, la corrección de la prueba sigue siendo larga y complicada (Kawarabayashi y Ozeki, 2010, p. 2619).

### 3.1 Grafo de caras internas de un Grafo Planar

Un grafo planar  $G$  tiene un conjunto de regiones cerradas no intersecadas  $F = \{f_1, \dots, f_k\}$  llamadas caras. Cada cara  $f_i \in F$  es representada por el conjunto de aristas que limitan su área interior. Toda arista  $\{u, v\}$  en  $G$  que no es la frontera de alguna cara de  $G$  se llama arista acíclica.

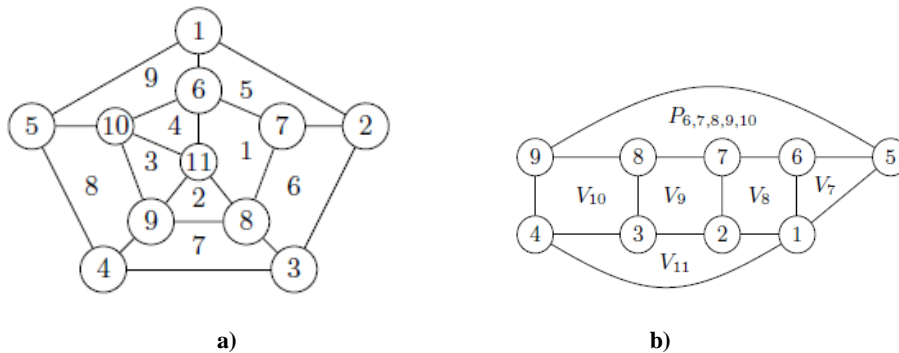
Dos caras  $f_i, f_j \in F(G)$  son adyacentes si tienen aristas comunes, esto es  $E(f_i) \cap E(f_j) \neq \emptyset$ . De lo contrario, son caras independientes. Dos aristas acíclicas son adyacentes si comparten un punto final común. Una arista acíclica es adyacente a una cara si tienen un solo vértice común. Un conjunto de caras es independiente si cada par de ellas son independientes. Construimos un grafo de caras internas  $G_f = (X, E(G_f))$  de  $G$ , de la siguiente manera:

1. Cada cara  $f_i$  tiene adjunto un nodo  $x \in V(G_f)$ .
2. Cada arista acíclica de  $G$  tiene adjunto un vértice de  $G_f$  etiquetado por la etiqueta de sus vértices.
3. Hay una arista  $\{u, v\} \in E(G_f)$  uniendo dos vértices adyacentes de  $G_f$ , cuando sus caras correspondientes (o aristas acíclicas) son adyacentes en  $G$ .

Se llama  $G_f$  al grafo de caras internas del grafo planar  $G$ . Notar que  $G_f$  no es el grafo dual de  $G$ , ya que en la construcción de  $G_f$  la cara externa no es considerada. El grafo de cara interna  $G_f$  de  $G$  proporciona un mapeo de la relación entre las caras de  $G$ , y es útil en la búsqueda de los patrones de grafos 3-coloreables. Notar que  $G_f$  también es un grafo planar, con vértices que representan caras o aristas de  $G$ , como podemos ver en la Figura 1 b).

Un patrón básico en el grafos planares es un grafo de ruedas que es un vértice único (llamado vértice centro de la rueda) adyacente a los vértices que forman un ciclo. Cada cara (llamado un eje de la rueda) es un triángulo. Hay dos clases de vértices en la rueda; el vértice centro y los vértices que forman el ciclo. Notar que una rueda de un grafo planar  $G$  es representado como un ciclo en su cara interna  $G_f$ . Las ruedas con un número de triángulos pares son 3-coloreables. Mientras tanto, cualquier grafo planar, conteniendo  $K_4$  o ruedas impares, requerirá 4 colores para colorear correctamente esos grafos. Sin embargo esas topologías no son los únicos casos 4-coloreables.

Extendemos la clase de ruedas considerando cualquier polígono como una cara de eje de la rueda. Este tipo de rueda se llama rueda poliédrica. Esto significa que hay vértices en el ciclo que rodea el vértice centro, que no son adyacentes al centro. Diferenciamos los vértices del ciclo en una rueda poliédrica como vértices de eje si son adyacentes al vértice centro; y como eje extra, cuando no son adyacentes al centro. Sin embargo, el vértice centro es un vértice común a toda la cara del eje de la rueda poliédrica.



**Fig. 1. a)** Grafo  $G$  con caras identificadas, **b)** Grafo de caras-internas  $G_f$  de  $G$ .

En el caso de las aristas en una rueda poliédrica, tenemos las aristas del ciclo y las aristas incidentes al centro, las cuales son llamadas aristas de radio.

Una rueda poliédrica de un grafo planar  $G$  es representada como un ciclo en su grafo de caras internas  $G_f$ . Sin embargo, un ciclo en  $G_f$  también puede codificar otro tipo de rueda de  $G$ . Una rueda de  $G$  donde el centro es un polígono en lugar de un solo vértice se llama un subgrafo poliédrico. Por ejemplo, en la Figura 1 b) hay 5 ruedas poliédricas cuyos centros están marcados por el único vértice centro de la rueda. También es un subgrafo poliédrico.

drico cuyo centro es el polígono formado por el conjunto de vértices  $\{6, 7, 8, 9, 10\}$ . Cualquiera de estas ruedas es una rueda impar si tiene un número impar de caras de ejes, de lo contrario, es una rueda par.

## 4 Grafos Planares

Es fácil (en tiempo lineal en el tamaño del grafo) reconocer si un grafo de entrada es 2-coloreable, ya que implica el reconocimiento de solo ciclos pares en el grafo. Similarmente se sabe por el teorema de Grotzsch's (1959) “que cualquier grafo planar libre de triángulos es 3-coloreable” (p. 109). Sin embargo, el reconocimiento del 3-coloreo de un grafo planar es un clásico problema NP-Completo. Es difícil reconocer entre los 3 o 4 colores de un grafo planar cuando este contiene triángulos, porque no es (al menos hasta ahora) una condición suficiente para reconocer la 3-colorabilidad de un grafo planar.

Sea  $Three = \{1,2,3\}$  el conjunto que contiene tres diferentes colores. Dado un grafo planar  $G$ , sea  $x_{v,c}$  la variable lógica denotando que el vértice  $v$  tiene el color  $c \in \{1,2,3,4\}$ . Para cada vértice  $v \in V(G)$  un conjunto  $Tabu(v)$  es asociado.  $Tabu(v)$  indica los colores prohibidos del vértice  $v$ . De hecho,  $Tabu(v)$  contiene las variables asociadas para los vértices en  $N(v)$  que ya han sido coloreados, es decir,  $Tabu(v) = \{c : (x_{u,c}), \text{ y } \{u,v\} \in E(G)\}$ .

Introducimos un coloreo típico para una rueda, donde su vértice centro se le asigna el primer color. Los colores 2, 3 son asignados de manera alterna para los vértices de ciclo. Este coloreo comienza en cualquier cara triangular de la rueda, y sigue en dirección opuesta de las manecillas del reloj desde los vértices del ciclo. Solo cuando el último vértice del ciclo es visitado se determina si el cuarto color es necesario.

En esta sección, empezamos nuestro análisis para buscar patrones básicos, donde es posible determinar si son 3 o 4 coloreables. Como las ruedas son un patrón básico general para formar grafos planares, empezamos considerando el coloreo de ruedas simples.

**Lema1.** La unión de ruedas simples pares, donde sus vértices centros son independientes, es 3-coloreable.

**Prueba.** El coloreo típico sobre las ruedas puede extenderse considerando la unión de las ruedas asignando el primer color a todos los vértices centro de las ruedas, ya que estos forman un conjunto independiente en el grafo. Aristas comunes entre las ruedas solo están dadas por las aristas del ciclo de las ruedas. Si los vértices centro son removidos del grafo, ya que estaban coloreados, el subgrafo restante es bipartito porque solo tiene las aristas de ciclos. Como solo quedan ciclos pares en subgrafo, entonces el subgrafo es 2-coloreable. Obtenemos un 3-coloreo para este tipo de grafos planares usando diferentes colores entre los vértices centro y los vértices del ciclo.

El siguiente lema propone un método para el 3-coloreo de componentes acíclicos, cuyos vértices tienen a lo más un color prohibido.

**Lema2.** Un componente acíclico, donde sus vértices tienen a lo más un color como restricción, es 3-coloreable.

**Prueba.** El componente acíclico es considerado como un árbol enraizado en  $v_r$ . Un coloreo en pre-orden está hecho de  $v_r$ , donde  $Color(v_r) = MIN\{Three-Tabu(v_r)\}$ . Si avanzamos en pre-orden para cada nuevo nivel a colorear, todo vértice y en el nuevo nivel tendrá a lo más 2 colores restringidos desde su nodo padre y el color que podría existir en  $Tabu(y)$ . Así, siempre está disponible un color de los 3 posibles en  $Three$ . El proceso del 3-coloreo finaliza, cuando todos los nodos del árbol han sido visitados en pre-orden.

**Lema3.** Sea  $A = \{f_1, f_2, \dots, f_n\}$  un conjunto de  $n$  caras no adyacentes, donde cada cara tiene al menos un vértice que no restringe el color 3, entonces  $A$  es 3-coloreable.

**Prueba.** Como  $A = \{f_1, f_2, \dots, f_n\}$  es un conjunto de caras no adyacentes, entonces algunas caras en  $A$  podrían compartir a lo más un vértice en común. Este lema se muestra por inducción en el número de caras en el conjunto.

1. Una sola cara es 3-coloreable, ya que cada ciclo con al menos un vértice sin restricción es 3-coloreable.
2. Supongamos que se mantiene la hipótesis en los conjuntos de hasta  $n-1$  caras.
3. Sea  $A$  un conjunto de  $n$  caras, donde cada cara tiene al menos un vértice que no es restringido con el color 3. Si hay una cara  $f_a \in A$  que es independiente (sin aristas comunes o vértices comunes) desde todas las otras caras de  $A$ , entonces  $f_a$  puede ser removido desde  $A$  ya que  $f_a$  es 3-coloreable (como en el caso 1). El conjunto restante en  $A$  tiene  $n-1$  caras, y se mantiene la hipótesis inductiva.

De lo contrario, las  $n$  caras en  $A$  comparten un vértice en común. Sea  $x \in f_i : \forall f_i \in A$  el vértice común. Si  $3 \notin Tabu(x)$ , luego asignando el color 3 a  $x$  y removiendo del grafo, todas las caras en  $A$  se abren y forman un grafo acíclico, que es 3-coloreable por el Lema2.

Asumiendo  $Tabu(x) = 3$ , pero  $\forall f_i \in A$ , esta  $y_i \in V(f_i)$  tal que  $Tabu(y_i) = \emptyset$ , por hipótesis. Asignando color 3 a cada uno de estos  $y_i$ 's, y eliminándolos de cada  $V(f_i)$ , se forma un componente acíclico, y este componente es 3-coloreable por el Lema2.

Cuando  $G_f$  es un árbol, decimos que  $G$  (su correspondiente grafo planar) es un *árbol poligonal* (López, De Ita y Neri, 2018, p. 93). Esto significa que, aunque  $G$  tiene ciclos, todos esos ciclos pueden estar arreglados como un árbol, cuyos nodos son polígonos en lugar de solo vértices de  $G$ . En este caso, un orden para visitar las caras del grafo planar proporciona un procedimiento eficiente para el 3-coloreo de  $G$ , como se afirma en el siguiente teorema.

**Teorema 1.** Si el grafo de cara interna de un grafo planar  $G$  tiene una topología de árbol, entonces  $G$  es 3 coloreable.

**Prueba.** Sea  $G_f$  el grafo de cara interna de un grafo planar  $G$ . Cada cara de  $G$ , representado por un nodo  $x \in V(G_f)$ , es 3-coloreable tal que es un ciclo simple. Además, toda arista acíclica de  $G$ , representada por un nodo de  $G_f$ , también es 3-colorable, ya que todo grafo acíclico es 2-coloreable. Proponemos un procedimiento 3-coloreable para  $G$  atravesando los nodos de  $G_f$  en pre-orden, colorear primero la cara del nodo padre de  $G_f$  y después, las caras de sus hijos. En cada nivel actual, se consideran las dos caras adyacentes (padre e hijo en  $G$ ). Ambas regiones tienen dos vértices comunes extremal  $x, y$  en sus fronteras comunes.

Estos vértices comunes se colorean primero, y luego, los vértices restantes, en ambas caras tienen dos colores prohibidos a lo más. Notar que no hay un par de vértices adyacentes  $u$  y  $v$  en cualquiera de las dos caras, tal que  $\{u, v\} \subseteq (N(x) \cap N(y))$ , porque entonces  $\{x, y, u, v\}$  forma un  $K_4$  y este subgrafo no puede ser parte de ningún árbol poligonal. Por lo tanto, para todos los vértices restantes en ambas caras, está disponible al menos un color de los tres posibles en *Three*. El proceso del 3-coloreo finaliza cuando todos los nodos del árbol de  $G_f$  se han visitado en pre-orden.

Si un grafo planar  $G$  no tiene una topología de árbol poligonal, significa que hay ciclos en  $G_f$  y, por lo tanto, ruedas en  $G$ . Para este tipo de grafos planares, es posible reconocer patrones gráficos 3-coloreables.

**Lema 4.** Toda rueda poliédrica es 3-coloreable.

**Prueba.** Si  $r_x$  es una rueda poliédrica, entonces hay una cara del eje que no es triangular. Por lo tanto, hay al menos un vértice  $v_e \in V(r_x)$  que no es adyacente al vértice centro  $v_x$  de la rueda, de lo contrario toda la cara del eje sería triangular. El grafo  $R_{-v_e} = (r_x - v_e)$  es un arreglo poligonal, y entonces es 3-coloreable por el Lema 3 y el Teorema 1. Cualquier típico 3-coloreo para  $R_{-v_e}$  puede extenderse a un 3-coloreo para  $r_x$  si  $v_e$  tiene el mismo color que  $v_x$ , porque en un coloreo típico el color del vértice centro no se usa en los vértices del ciclo. Por lo tanto, el color del centro no se ha utilizado para los vértices en  $N(v_e)$ .

La unión de ruedas 3-coloreables no es necesariamente 3-coloreable. Por ejemplo, cada rueda individual en el grafo de la Figura 2 b) es 3-coloreable, cuando la cara común se asume como parte de cada rueda, y por el Lema 4, cada una de ellas es 3-coloreable. Sin embargo el grafo final (unión de las ruedas) es 4-coloreable como veremos en la siguiente sección.

#### 4.1 Método para determinar la 3 o 4 colorabilidad de una secuencia de ruedas

Proponemos un método lógico en orden para reconocer entre 3 o 4 colores para la colorabilidad de la unión de ruedas. Sea  $G$  un grafo resultante de la unión de ruedas simples compartiendo caras comunes. Nuestra propuesta se basa en la construcción de una fórmula proposicional  $F_G$  expresada en forma normal conjuntiva, y cuya Satisfactibilidad codifica la 3-colorabilidad de  $G$ . De lo contrario,  $G$  sería 4-coloreable.

Para nuestra propuesta, consideremos primero un par de ruedas simples  $r_x$  y  $r_y$  que comparten una cara común que es denotada como  $(r_x \cup_f r_y)$ , donde  $f = (F(r_x) \cap F(r_y))$ . Los vértices centro de las ruedas son etiquetados por  $x$  para  $r_x$ , y  $a$  para  $r_y$ , como se ilustra en la Figura 2 a). Una tripleta de variables simbólicas  $\{x, y, z\}$  es asociado a  $V(r_x)$ , mientras la tripleta de variables simbólicas  $\{a, b, c\}$  es asociado a  $V(r_y)$  de la siguiente manera.

1. La variable  $x$  es asociada al vértice centro de  $r_x$ , mientras la variable  $a$  es asociada al vértice centro de  $r_y$ .
2. Los vértices del ciclo de  $r_x$  son asociados con las variables  $y$  y  $z$  de manera alterna. Comienza con una cara adyacente triangular a la cara común  $f$  y siguen las otras caras triangulares en  $r_x$  hasta que todos los vértices del ciclo estén cubiertos.

3. Los vértices del ciclo de  $r_y$  son asociados con las variables  $b$  y  $c$  de manera alterna. Comienza con una cara adyacente triangular a la cara común  $f$  y siguen las otras caras triangulares en  $r_x$  hasta que todos los vértices del ciclo estén cubiertos.

La topología de  $(r_x \cup_f r_y)$  define el tipo de restricciones formando  $F_G$  de la siguiente manera.

- I) Los vértices  $V(f)$ , que son vértices comunes entre  $V(r_x) \cap V(r_y)$ , determinan las restricciones de igualdad entre sus variables correspondientes:  $(y \oplus z) = (a \oplus b)$ , donde  $(\oplus)$  denota el operador lógico *xor*. Esto significa que los vértices en  $V(r_x) \cap V(r_y)$  define colores iguales (variables con mismos valores).
- II) Una arista  $e \in E(G)$ , con puntos finales en  $V(r_x)$  y  $V(r_y)$ , define una restricción de desigualdad entre sus variables correspondientes:  $(y \oplus z) \neq (a \oplus b)$ . Esto significa que vértices adyacentes en  $V(f)$  define diferentes colores (variables con diferentes valores).
- III)  $F_G$  también considera que cualquier par de vértices debe tener colores diferentes. Esto se codifica como:  $(x \neq y) \wedge (x \neq z) \wedge (z \neq y) \wedge (a \neq b) \wedge (b \neq c) \wedge (c \neq a)$ .
- IV) En  $F_G$  también se agregan las restricciones definiendo que todo vértice en  $V(r_x) \cup V(r_y)$  debe de tener uno de los tres posibles colores y se codifica como:  $((x = a) \vee (x = b) \vee (x = c)) \wedge ((y = a) \vee (y = b) \vee (y = c)) \wedge ((z = a) \vee (z = b) \vee (z = c))$ .

Notar que el tipo de restricción I y II son cláusulas unitarias, y que cualquier desigualdad puede considerarse como la negación de una restricción de igualdad. Además, un par de cláusulas unitarias contradictorias implica la insatisfactibilidad de  $F_G$ . La Satisfactibilidad de  $F_G$  determina que  $(r_x \cup_f r_y)$  es 3-coloreable. De otra manera,  $(r_x \cup_f r_y)$  es 4-coloreable. Cuando  $F_G$  es satisfactible, entonces una función biyectiva  $f_R: \{x, y, z\} \rightarrow \{a, b, c\}$  existe. Esto determina la existencia de un 3-coloreo válido para los vértices en el par de ruedas.

La Figura 2 a) y b) ilustran los dos casos diferentes para los valores de  $F_G$ . Para el grafo en la Figura 2 a), las restricciones tipo II define que  $(b \neq y) \wedge (a \neq y)$ , y entonces  $(c = y)$  se infiere debido a las restricciones tipo IV, ya que es validado por la única restricción tipo I. Ya que no hay más restricciones tipo II, entonces la fórmula  $F_G$  puede ser satisfactible por las asignaciones  $((a = x) \wedge (b = z))$ , o  $((a = z) \wedge (b = x))$ . Por lo tanto, el grafo de la Figura 2 a) es 3-coloreable. Un 3-coloreo válido es, por ejemplo,  $(c = y) \wedge (a = x) \wedge (b = z)$ , donde  $\{a, b, c\}$  representa cualquier permutación de los valores  $\{1, 2, 3\}$ .

Nuestra propuesta para revisar la 3-colorabilidad de una secuencia de ruedas pares funciona cuando los centros de las ruedas no son independientes, o cuando el grafo está formado por la unión de más de dos ruedas.

Por ejemplo, consideremos el grafo de la Figura 2 b). En este caso, reconocemos tres ruedas y, por lo tanto, las tripletas  $\{x_1, y_1, z_1\}, \{x_2, y_2, z_2\}, \{x_3, y_3, z_3\}$  son asociados a  $V(r_1)$ ,  $V(r_2)$ , y  $V(r_3)$ , respectivamente. Las restricciones tipo I determinan las igualdades:  $(y_3 = z_1) \wedge (z_3 = y_2) \wedge (z_3 = y_1)$ . Mientras tanto, las restricciones tipo II determinan la desigualdad  $(y_1 \neq y_2)$ . No es difícil inferir  $(y_1 = y_2)$  de las condiciones  $(z_3 = y_2) \wedge (z_3 = y_1)$ . Sin embargo, esta igualdad inferida contradice la desigualdad tipo II:  $(y_1 \neq y_2)$ . Por lo tanto,  $F_G$  es insatisfactible. Como  $F_G$  es insatisfactible, entonces el grafo planar no es 3-coloreable y, por lo tanto,

es necesario usar un cuarto color para colorear correctamente el grafo, de acuerdo con el teorema de 4CT.

Notar que la revisión de la Satisfactibilidad de  $F_G$  se puede hacer en tiempo polinomial en el tamaño del grafo  $G$ . Esto es posible, ya que consiste en la aplicación de resolución unitaria de las restricciones Tipo I y Tipo II contra todas las restricciones en  $F_G$  así como la aplicación de la transitividad de las restricciones de igualdad. Si hay una contradicción en  $F_G$ , entonces se generará durante el proceso de resolución unitaria, implicando que es necesario 4 colores para colorear  $G$ .

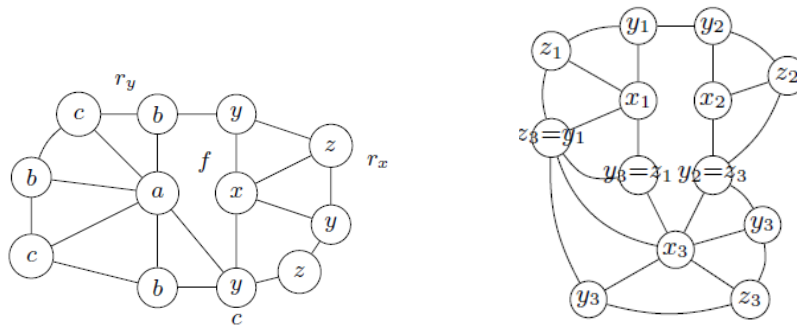


Fig. 2. a) Una unión de ruedas 3-coloreable, b) Una unión de ruedas 4-coloreable

## 5 Conclusiones

Reconocemos los patrones básicos de grafos, formando grafos planares, como ruedas con diferentes tipos de centros o caras de eje. Analizamos para estas ruedas la posibilidad de determinar su 3 o 4 colorabilidad. También consideramos diferentes composiciones entre ruedas, y analizamos su proceso de colorabilidad. Proponemos un método eficiente basado en la construcción de una forma normal conjuntiva  $F_G$ , que es formada por las restricciones de igualdad y desigualdad definidas por las relaciones de los vértices en una secuencia de ruedas en  $G$ .

## Referencias

- Cortese P., M. Patrignani (2004), Planarity Testing and Embedding, Press LLC (p.1-39).  
Dvorák, Z., Král, D., Thomas, R. (2009), *Three-coloring triangle-free graphs on surfaces*, Proc, 20th ACM-SIAM Symp. on Discrete Algorithms (pp. 120-129).  
Fomin F. V. and Kratsch D.(2010), Exact exponential algorithms, Texts in Theoretical Computer Science, an EATCS Series, Springer.  
Golumbic M. C. (2004), Algorithmic Graph Theory and Perfect Graphs, 2nd ed. North Holland.

- Grötzsch H. (1959), Ein Dreifarbensatz für dreikreisfreie Netze auf der Kugel, *Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math.-Natur. Reihe* 8 (pp. 109-120).
- Harary F. (1969), *Graph Theory*. Addison-Wesley, Reading, Mass.
- Kuratowski K. (1930), Sur le probleme des courbes gauches en topologie, *Fund. Math.*, 15 (pp. 271-283).
- Kawarabayashi K., Ozeki K. (2010), A simple algorithm for 4-coloring 3-colorable planar graphs, *Theoretical Computer Science* 411, (pp. 2619-2622), Elsevier.
- López C., De Ita G., Neri A. (2018), A. Modelling 3-Coloring of polygonal trees via Incremental Satisfiability. LNCS 10880, (pp. 93-104), Springer Verlag.
- Lozin V. V. and Malyshev D. S. (2017), Vertex coloring of graphs with few obstructions, *Discrete Applied Mathematics*, vol. 216, (pp. 273-280), Elsevier.
- Mertzios, G.B., Spirakis, P.G (2013), Algorithms and almost tight results for 3-colorability of small diameter graphs (pp. 332-343). Technical report, [arxiv.org/pdf/1202.4665v2.pdf](https://arxiv.org/pdf/1202.4665v2.pdf).
- Robertson N., D.P. Sanders D.P., P.D. Seymour, and R. Thomas (1997), The four color theorem. *J. Combin. Theory Ser. B*, 70:24, (pp. 2-44).
- Stacho, J. (2010), “3-colouring AT-free graphs in polynomial time”. en Cheong, O., Chwa, K.-Y., Park, K. (eds.) *ISAAC 2010*. LNCS, vol. 6507 (pp. 144-155). Springer, Heidelberg. [https://doi.org/10.1007/978-3-642-17514-5\\_13](https://doi.org/10.1007/978-3-642-17514-5_13).
- Wagner K. (1937), Über eine Eigenschaft der ebenen Komplexe. *Mathematische Annalen*, 114 (pp. 570–590), Springer Verlag.



# Capítulo 2

## Diseño de recursos para la evaluación de una ontología de dominio pedagógico

Yuridiana Alemán, María Somodevilla, Darnes Vilariño

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

{yuridiana.aleman, mariajsomodevilla, dvilarinoayala}@gmail.com

**Resumen.** El presente trabajo muestra algunos avances de la aplicación de una metodología para el aprendizaje ontológico en un subdominio pedagógico. La propuesta general se centra en el uso de técnicas de Procesamiento de Lenguaje Natural y Recuperación de Información, para el idioma español en todas las fases del aprendizaje ontológico. Particularmente, se analizan dos recursos para la fase de evaluación del proceso: Una ontología manual y la creación de un conjunto de validación de datos (*gold standard*). Se trabajó con tres clases predefinidas con la ayuda de un experto del dominio, dichas clases están representadas con artículos en español publicados en revistas de pedagogía o psicología.

**Palabras Clave:** Aprendizaje ontológico, ontología, corpus pedagógico.

### 1 Introducción

Una ontología es una formalización explícita, en donde un texto se representa por medio de relaciones semánticas y palabras clave. Existen varias vertientes de esta estructura, siendo las más comunes las ontologías de dominio, las cuales son generalizaciones sobre tareas específicas en un área concreta del conocimiento. En la actualidad, las ontologías se utilizan en varias áreas de la inteligencia artificial y en otros dominios del conocimiento.

Tradicionalmente, el proceso de construcción de una ontología requiere una interacción muy alta con los expertos de dominio, esto origina tiempo para la creación de la ontología, sobre todo cuando el dominio es extenso (Faria y Girardi, 2014). Aunque se han publicado trabajos para una creación ontológica automática o semiautomática; estas investigaciones involucran una ontología manual o la creación de un corpus con características muy específicas para extraer los datos.

El objetivo general de la investigación consiste en la construcción de ontologías de manera semiautomática a partir de textos pedagógicos no estructurados para expresar con claridad y precisión el conocimiento. Este artículo se enfoca en la fase de evaluación del proyecto, particularmente en el análisis de los elementos necesarios para la evaluación de los experimentos realizados a lo largo del proyecto: la creación de un conjunto de datos para validación (*gold standard*). Para el dominio de estudio se utilizan tópicos relacionados con la implementación de técnicas de enseñanza dentro del salón de clases; por lo tanto, se toman en cuenta las estrategias de enseñanza aprendizaje, los estilos de aprendizaje y tipos de inteligencia.

La estructura del artículo se muestra a continuación: La sección 2 desarrolla los elementos teóricos relacionados con el dominio y la construcción de ontologías. En la sección 3 se muestran las investigaciones más recientes, además, se anexan algunos trabajos relacionados con el dominio pedagógico, posteriormente la sección 4 explica la metodología general. Finalmente, la sección 5 analiza los resultados parciales obtenidos y en la sección 6 se muestran las conclusiones obtenidas hasta el momento.

## 2 Conceptos teóricos

En las ciencias computacionales, una ontología se define como “una especificación formal de una conceptualización” (Gruber, 1995, p. 1). Otra definición es la de Weigand (1997), el cual la define como “una base de datos que describe los conceptos en el mundo o algún dominio, algunas de sus propiedades y como los conceptos se relacionan entre sí” (p. 1).

Para el proceso de creación de una ontología, Mizoguchi et al. (1997) propone los siguientes pasos:

1. Definir el dominio y el alcance de la ontología, la aplicación final y a qué tipo de preguntas responderá dicha ontología.
2. Considerar el rehusó de ontologías. En caso de poder aplicarse, se recolectan términos y características de ontologías comunes ya existentes.
3. Enumerar términos importantes en la ontología. Definir una lista de los términos que se quieren almacenar en la ontología, así como de sus propiedades y las relaciones que existen entre ellos.
4. Definir las clases y la jerarquía de las clases (estructura conceptual del dominio). Primero se crean las clases para los conceptos generales en el dominio.
5. Definir las propiedades de las clases, esto es, la estructura interna de los conceptos.
6. Crear instancias de clases individuales en la jerarquía.

Dentro del dominio pedagógico, se establecieron tres clases principales a fin de obtener una herramienta de apoyo para el docente en clases presenciales. Las clases se describen a continuación:

**Estilos de aprendizaje:** Reflejan la forma en que el individuo aprende y describen las condiciones bajo las que un discente se encuentra en la mejor situación para aprender. Para esta investigación se analiza el modelo de David Kolb, el cual se basa en los procesos psicológicos de percepción y procesamiento (Olivos, et al. 2016) para proponer 4 estilos de aprendizaje: activo, reflexivo, pragmático y teórico.

**Tipos de inteligencias:** Según Gardner, la inteligencia “implica la habilidad necesaria para resolver un problema o para elaborar productos que son importantes en un contexto cultural” (2001, p. 10). Los seres humanos poseen una gama de capacidades y potenciales que se pueden emplear de muchas maneras productivas, esta idea da origen a las inteligencias múltiples de Gardner, el cual distingue ocho tipos de inteligencias: lógico-matemática, lingüística, espacial, musical, corporal, intrapersonal, interpersonal y naturalista.

**Estrategias de enseñanza aprendizaje:** Según Genovard y Gotzens (1990), las estrategias de aprendizaje son comportamientos que el estudiante desarrolla durante su proceso de aprendizaje, influyendo en su proceso de codificación de la información. Su aplicación implica una secuencia de actividades, operaciones o planes dirigidos a la consecución de metas de aprendizaje. Aunque existen muchos enfoques para la clasificación de las estrategias de aprendizaje, Gonzalez y Tourón (1992) mencionan tres tipos principales: cognitivas, metacognitivas y las estrategias de manejo de recursos.

### 3 Estado del arte

Existen investigaciones enfocadas al dominio pedagógico, la mayoría de ellas con un enfoque manual. La investigación de Méndez, Carranza y Ocampo (2015) presentan una modelación ontológica para la personalización del aprendizaje virtual basada en la teoría de las inteligencias múltiples de Gardner. Se propone creación manual y se evalúa por medio de consultas realizadas por expertos en el dominio. Autores como Uskov, et al. (2016) se centran en el aprendizaje autónomo dentro del aula con la ayuda de la tecnología tomando como referencia los tipos de inteligencia de los estudiantes para esto proponen una ontología en base al Internet de las cosas (IoT).

Ameen, et al. (2012) explica el proceso de creación de una ontología a partir de la información de cursos que se ofertan en nivel superior, donde el estudiante puede elegir los cursos a tomar de acuerdo a su historial académico. Otras investigaciones como la de Hu, et al. (2016) y recientemente Hssina, et al. (2017) se centran en la educación en línea, donde las ontologías se definen manualmente a partir de recursos XML disponibles en Internet, y la evaluación también es un proceso manual.

Es importante mencionar que antes de iniciar el proceso de extracción de elementos principales, se debe tener un corpus para el dominio a trabajar, por lo que se analizaron algunas investigaciones sobre la construcción de corpus en diferentes dominios. Grljevic y Bosnjak (2015) se centran en la creación del corpus lingüístico relevante escrito en lengua serbia, en dicha investigación, el enfoque es el análisis de sentimientos de los contenidos

generados por los estudiantes en la educación superior. Teixeira et al. (2011) se analizaron el problema de crear un corpus de referencia para la clasificación de artículos de noticias en escenarios de etiquetas múltiples. Los autores proponen un enfoque semiautomático para crear un corpus de referencia que utiliza tres métodos de clasificación auxiliares: máquinas de vectores de soporte, clasificadores de vecinos más cercanos y otro basado en un diccionario.

Respecto a la evaluación de ontologías algunos métodos en la literatura son los clásicos en la recuperación de información, los cuales involucran dos métricas estándar para valorar la ontología: precisión definida por la proporción de material relevante recuperado, y recuerdo que se define como capacidad de la ontología para recuperar objetos. Otros métodos están basados en diferentes teorías matemáticas (Senso et al., 2011), analizan fundamentalmente, la profundidad de la descripción en función del número de clases principales, subclases, notaciones formales y profundidad media del árbol taxonómico.

Por su parte, Raad y Cruz (2015) dividieron las técnicas de evaluación en cuatro grupos, de acuerdo a los métodos utilizados en el proceso y el enfoque de las ontologías. El enfoque basado en *gold standard* es el más directo, ya que permite obtener métricas de recuperación de información para evaluar la efectividad de los experimentos realizados. El enfoque basado en corpus se usa para evaluar el grado en que una ontología cubre un dominio dado, mientras que el enfoque basado en tareas intenta medir hasta qué punto una ontología ayuda a mejorar los resultados de una determinada tarea. Finalmente, el enfoque basado en criterios mide hasta qué punto una ontología o taxonomía se adhiere a ciertos criterios deseables.

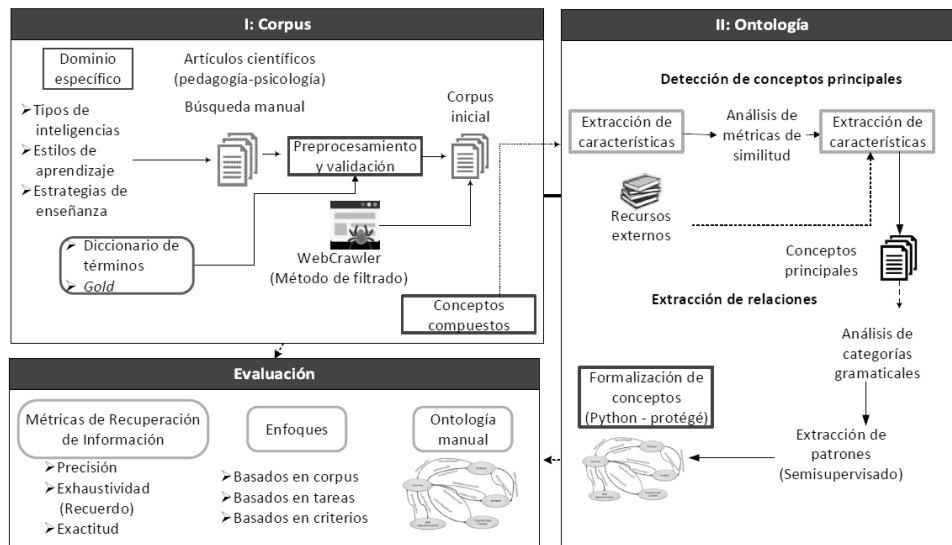
## 4 Metodología

La Figura 1 muestra la metodología general para la investigación, la cual consta de tres fases principales. El primer paso consiste en la creación de un corpus compuesto de artículos publicados en el idioma español, dentro de esta fase se realizó un proceso de validación de clases y expansión de un corpus inicial a fin de incrementar la riqueza del vocabulario. En la segunda fase se obtienen los conceptos principales y las relaciones taxonómicas y no taxonómicas para la obtención de la ontología. Finalmente, se anexa un módulo de evaluación para determinar la eficacia de los experimentos, dentro del cual se necesitarán las herramientas analizadas en el presente artículo.

El corpus final está conformado por artículos relacionados con los tópicos principales seleccionados anteriormente (tipos de inteligencias, estrategias de enseñanza y estilos de aprendizaje), de tal manera que se crean tres clases iniciales. Para la detección de conceptos cada una de las clases se procesa de manera independiente.

Los avances previos de esta investigación integran la propuesta de un método para la detección de conceptos principales basado en métricas de similitud textual, un algoritmo

de filtrado de documentos para la expansión del corpus inicial y el análisis de patrones de categorías gramaticales para la detección de relaciones entre conceptos.



**Fig. 1.** Metodología General de la investigación

La Tabla 1 muestra el número de artículos por clase, así como el total de palabras y vocabulario. La clase estilos de aprendizaje tiene un mayor número de artículos, pero un menor vocabulario mientras que la clase tipos de inteligencias presenta un vocabulario mayor con sólo 61 artículos.

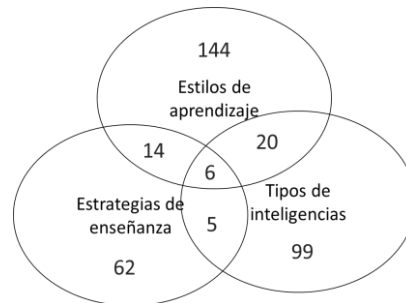
**Tabla 1.** Número de artículos y vocabulario por clase.

Clase	Artículos	Palabras	Vocabulario
Estilos de aprendizaje	83	215,211	8,129
Estrategias de enseñanza	63	217,569	8,817
Tipos de inteligencias	61	198,452	9,474

Analizando estos datos desde el punto de vista del dominio, la teoría de las inteligencias múltiples manejan 8 tipos de inteligencias y las estrategias de enseñanza analizan varios tipos y subtipos. En cambio, los estilos de aprendizaje se clasifican en 4, lo que hace que los artículos de esta clase sean más similares en cuanto a vocabulario.

## 5 Conjunto de datos para validación

El conjunto de validación se generó manualmente con la ayuda del experto en el dominio y contiene una serie de conceptos por clase considerados como importantes. Este recurso cuenta con conceptos formados por una o dos palabras. Los conceptos formados por dos palabras son los que originan la necesidad de integrar técnicas que detecten colocaciones en los experimentos previamente implementados.



**Fig. 2.** Número de elementos del conjunto de evaluación por clase

Las listas de conceptos importantes contienen 130 elementos para los tipos de inteligencias, 87 para las estrategias de enseñanza y 184 para los estilos de aprendizaje, sin embargo, estas listas tienen elementos en común. La Figura 2 muestra el diagrama de Venn con el número de elementos que se comparten por clase.

**Tabla 2.** Elementos del conjunto de evaluación por clase (extracto)

<i>Estilos de aprendizaje</i>	<i>Estrategias de enseñanza</i>	<i>Tipos de inteligencia</i>	<i>Común a 3 clases</i>
Tic	Aprendizaje	Persona	Docente
Reflexivo	Estrategia	Inteligencia múltiple	Estrategia
Estudio	Conocimiento	Desarrollo	Estudiante
Pragmático	Metacognitiva	Teoría	Habilidad
Académico	Análisis	Gardner	Individuo

Las tres clases comparten 6 conceptos, mientras que la clase de estilos de aprendizaje es la que comparte un mayor número de elementos con otras clases (14 con estrategias de enseñanza y 20 con tipos de inteligencias). La clase estrategias de enseñanza, a pesar de tener más texto que las demás, tiene menos conceptos importantes y comparte pocos con otras clases. La tabla 2 contiene ejemplos de algunos conceptos del conjunto de evaluación por clase y los que están presentes en los tres conjuntos.

## 6 Conclusiones y trabajo futuro

Como resultados previos en la investigación general, se tienen los siguientes avances:

- Creación de un corpus en español con tres clases, el cual está constituido por artículos publicados en revistas de ciencias sociales. Se realiza una validación por medio de técnicas de agrupamiento automático, las cuales arrojan resultados similares al agrupamiento realizado de manera manual.
- Expansión del corpus inicial mediante un proceso automático, incrementando de 51 a 207 el número de instancias del corpus.
- Propuesta para la extracción automática de colocaciones a fin de analizar conceptos compuestos en el corpus.
- Obtención de una lista de conceptos relacionados con cada una de las clases principales mediante métricas de similitud textual. Primero se obtienen con un *rate* fijo y posteriormente con el diseño de un experimento para encontrar el *rate* óptimo para cada una de las clases.
- Experimentos para la extracción de relaciones basados en la extracción de patrones y análisis de categorías gramaticales.

En cuanto a los resultados presentados en este trabajo, se presentan dos herramientas que serán de utilidad para la evaluación de las metodologías y procedimientos realizados hasta el momento, especialmente aquellos enfocados en la detección de clases y relaciones. El conjunto de validación cuenta con elementos importantes para la realización semiautomática de ontologías, mientras que la creación manual de dichas ontologías, permite evaluar los métodos para la extracción de relaciones entre conceptos. Como trabajo futuro se realizarán consultas en la ontología creada, además de corroborar el conjunto de evaluación con las estructura del corpus.

## Referencias

- Ameen, A., Khan, K. U. R., & Rani, B. P. (2012). Creation of ontology in education domain. *2012 IEEE Fourth International Conference on Technology for Education*, pp. 237–238
- Faria, C., Girardi, R. (2014). “A domain-independent process for automatic ontology population from text”, *Science of Computer Programming. Special Issue on Systems Development by Means of Semantic Technologies*, Vol 95, Part 1, pp. 26-43.
- Gardner, H. (2001). *Estructuras de la Mente*. Fondo de Cultura Económica.
- Genovard, C. & Gotzens, C. (1990). *Psicología de la instrucción*. Santillana
- González, M. & Tourón, J. (1992). *Autoconcepto y rendimiento escolar: sus implicaciones en la motivación y en la autorregulación del aprendizaje*. Eunsa

- Grljevic, O. & Bosnjak, Z. (2015). Development of serbian higher education corpus. *2015 16th IEEE International Symposium on Computational Intelligence and Informatics (CINTI)*. pp. 177–181
- Gruber, T. (1995). “Toward principles for the design of ontologies used for knowledge sharing”, *International Journal of Human-Computer Studies - Special issue: the role of formal ontology in the information technology*, Vol 43, pp. 907 - 928.
- Hssina, B., Bouikhalene, B., & Merbouha, A. (2017). An ontology to assess the performances of learners in an e-learning platform based on semantic web technology: Moodle case study. *Europe and MENA Cooperation Advances in Information and Communication Technologies* pp. 103–112
- Hu, J., Li, Z., & Xu, B. (2016). An approach of ontology based knowledge base construction for chinese k12 education. *2016 First International Conference on Multimedia and Image Processing (ICMIP)*, pp. 83–88
- Méndez, N. D. D., Carranza, D. A. O., & Ocampo, M. G. (2015). Representación ontológica de perfiles de estudiantes para la personalización del aprendizaje. *Revista Educación en Ingeniería*, Vol. 10(19), pp. 105–115.
- Mizoguchi, R., Ikeda, M., & Sinita, K. (1997). Roles of shared ontology in ai-ed research. *Proc. of AI-ED*, Vol 97, (pp. 537–544).
- Olivos, P., Santos, A., Martín, S., Cañas, M., Gómez, E., & Maya, Y. (2016). The relationship between learning styles and motivation to transfer of learning in a vocational training programme. *Suma Psicológica*, Vol. 23(1), pp.25–32.
- Raad, J. & Cruz, C. (2015). A survey on ontology evaluation methods. *Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2015*, pp. 179–186.
- Senso, J. A., Leiva-Mederos, A., & Dominguez-Velasco, S. (2011). Modelo para la evaluación de ontologías, aplicación en Onto-Satcol. *Revista Española de Documentación Científica*, Vol. 34(3), pp. 334–356.
- Teixeira, J., Sarmento, L., & Oliveira, E. (2011). Semi-automatic creation of a reference news corpus for fine-grained multi-label scenarios. *6th Iberian Conference on Information Systems and Technologies (CISTI 2011)*, pp. 1–7.
- Uskov, V., Pandey, A., Bakken, J. P., & Margapuri, V. S. (2016). Smart engineering education: The ontology of internet-of-things applications. *2016 IEEE Global Engineering Education Conference (EDUCON)*, pp. 476–481.
- Weigand, H. (1997). “A multilingual ontology-based lexicon for news filtering-the TREVI project”, *Proceedings of the IJCAI Workshop on Multilingual Ontologies-Nagoya*.



# Capítulo 3

## Etiquetado de video e identificación de unidades de acción para el reconocimiento de emociones centradas en el aprendizaje

Yesenia Nohemí González-Meneses, Josefina Guerrero-García

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

yeseniaglez0@gmail.com, joseguga01@gmail.com

**Resumen.** En el presente trabajo se describen las actividades realizadas para identificar unidades de acción de dos emociones centradas en aprendizaje: aburrido e interesado. Emociones de estudiantes grabadas en video durante el uso de un software para aprender algebra básica. Los videos fueron etiquetados de manera automática bajo las estrategias de etiquetado humano y auto etiquetado. Con los videos etiquetados se plantearon estrategias para acelerar el análisis de datos y poder seleccionar y fusionar características para posteriormente entrenar algoritmos de reconocimiento de patrones que discriminen entre las dos emociones mencionadas.

**Palabras Clave:** Etiquetado, unidades de acción (*AUs*), emociones centradas en el aprendizaje.

### 1 Introducción

El trabajo presentado en este artículo resume las actividades que se están llevando a cabo como parte del proyecto de investigación que intenta identificar emociones centradas en el aprendizaje (aburrido e interesado) de forma automática mediante algoritmos de aprendizaje computacional. Se inicia con videos capturados mientras estudiantes realizan una actividad de aprendizaje a través del uso de software educativo. Estos videos son etiquetados bajo dos esquemas diferentes: 1) Etiquetado realizado por observadores entrenados bajo la teoría de Paul Ekman (2002) usando su "*Manual del Sistema de codificación facial*". A través de la observación directa de los videos se realiza el etiquetado manual del video completo bajo el criterio del observador; 2) Etiquetado considerando las repuestas que cada estudiante de manera personal asigna mediante test a su estado emocional en tres momentos diferentes de tiempo durante la ejecución del experimento (aproximadamente 40 minutos). Con las emociones seleccionadas por cada alumno se etiqueta el minuto de tiempo en el cual asignó su respuesta. El etiquetado es la

referencia que se utiliza para clasificar las imágenes en algunos de los estados emocionales de nuestro interés (interesado y aburrido). Las características referentes a las *AUs* de cada imagen clasificada serán los datos de entrenamiento de algoritmos de aprendizaje automático. Para poder utilizar estos datos, las imágenes son procesadas para después hacer una extracción y selección de características. Se selecciona un determinado número de fotogramas por video y cada uno pasa por un proceso que mejora la imagen, para después segmentar la cara del estudiante, identificar los puntos de referencia facial y obtener un conjunto de características faciales de las cuales solo un conjunto de ellas es seleccionado como datos de entrada a un algoritmo de clasificación fuzzy de *AUs*. Al final se obtienen valores de membresía para 27 *AUs* por fotograma. Estos valores son los datos de entrada para el entrenamiento de algoritmos de clasificación. La clasificación, de acuerdo con las unidades de acción presentadas en cada emoción, es la siguiente etapa dentro del proyecto de investigación mencionado con anterioridad.

## 2 Preliminares

Las expresiones faciales son de los canales de comunicación no verbal más importante para expresar las emociones internas y las intenciones. Ekman (2002) define seis expresiones básicas (angustia, disgusto, miedo, felicidad, tristeza y sorpresa) que son universales entre los seres humanos. El desarrollo de sistemas para reconocimiento de expresiones faciales (FER, *Facial Expression Recognition*) automatizado ha presentado varios avances, muchos de ellos muestran comportamiento inadecuado en aplicaciones prácticas o carencia de generalización debido a las condiciones controladas en las cuales fueron desarrollados. El reconocimiento de expresiones es un proceso dividido en tres fases: inicio, ápice y culminación, el ápice describe la expresión en su máxima intensidad y la culminación describe la expresión desvaneciéndose. La mayoría de las veces la entrada del evento de la expresión facial desde el inicio hasta su culminación es muy rápido, lo cual hace a el proceso de reconocimiento muy complejo. Aunque el reconocimiento de expresiones faciales puede ser conducido usando múltiples sensores, uno de los enfoques más estudiados es a través del análisis de imágenes faciales visibles debido a que la expresión visual es uno de los principales canales de información en la comunicación interpersonal. Una cámara es el sensor más prometedor porque provee información clave para el FER y no es invasivo.

El FER automático es una tecnología que usa marcadores biométricos y de apariencia para detectar emociones en caras humanas. Los enfoques más utilizados para el FER automático se dividen en dos grupos en base a si las características son obtenidas manualmente o son generadas a través de la entrada de datos a una red neuronal profunda.

El enfoque convencional el FER está compuesto por tres pasos principales: 1) detección de la cara, 2) extracción de características, y 3) clasificación de la expresión. Primero, la cara es segmentada desde una imagen de entrada, y los puntos de referencia y los componentes faciales (como, ojos, nariz, boca) son detectados. Segundo, varias

características espaciales y temporales son extraídas de los componentes faciales. Tercero, los clasificadores de FE pre entrenados, tales como máquina de vectores de soporte (SVM, *Support Vector Machine*), *AdaBoost* o árboles de decisión, producen los resultados de reconocimiento usando las características extraídas.

En contraste al enfoque tradicional que usa características extraídas manualmente, surge el aprendizaje profundo como un enfoque general para el aprendizaje automático, produciendo resultados en el estado del arte en muchos estudios de visión por computadora con la disponibilidad de grandes datos. El FER basado en aprendizaje profundo reduce en gran medida la dependencia de los modelos basados en la cara física y otras técnicas de procesamiento al permitir que el aprendizaje ocurra “de extremo a extremo” directamente desde las imágenes de entrada (Walecki et al. 2017). Entre los diferentes modelos de aprendizaje profundo disponibles, la red neuronal convolucional (CNN, *convolutional neural network*), es el modelo de red más popular. En los enfoques basados en la CNN, la imagen de entrada es convolucionada a través de una colección de filtros en las capas convolucionales para producir un mapa de características. Luego, cada mapa de características es combinado con redes completamente conectadas y la expresión facial es reconocida como perteneciente a una clase particular basada en la salida de un algoritmo.

El FER también se divide en dos grupos de acuerdo a si usa fotogramas o imágenes de video (Walecki et al., 2017). El primero, el FER estático (basado en fotogramas) se basa únicamente en características faciales estáticas obtenidas por la extracción manual de fotogramas seleccionados de una secuencia de imágenes. El segundo, FER dinámico (basado en video) utiliza características espaciotemporales para capturar la expresión dinámica en secuencias de expresiones faciales. Para los sistemas de FER automáticos, varios tipos de enfoques convencionales han sido estudiados. Lo común en estos enfoques es detectar la región de la cara y extraer características geométricas, características de apariencia o un híbrido de características geométricas y de apariencia sobre la cara etiquetada.

Para las características geométricas, la relación entre componentes faciales es usada para construir un vector de características para entrenamiento. Por ejemplo, en (Ghimire y Lee, 2013) usaron dos tipos de características geométricas basadas en la posición y ángulo de 52 puntos de referencia faciales. Para la clasificación, dos métodos son presentados, uno usando *AdaBoost* multiclase y otro usando una máquina de vectores de soporte (SVM) sobre los vectores de características.

Las características de apariencia son usualmente extraídas de la región global de la cara o de diferentes regiones que contienen diferentes tipos de información. Un enfoque que define un tipo de características de apariencia es el sistema de codificación de acción facial (FACS). Este es un sistema basado en cambios de músculos faciales y puede caracterizar acciones faciales para expresar emociones humanas individuales. FACS codifica los movimientos de músculos faciales específicos llamados unidades de acción (*AUs*, Action Units), las cuales reflejan distintos cambios momentáneos en la apariencia facial (Hjortsj et al, 2019). Las unidades de acción facial (*AUs*) codifican las acciones fundamentales (46 *AUs*) de un músculo o de un grupo de músculos típicamente vistos

cuando la expresión facial de una emoción en particular es producida. Para reconocer la emoción facial, la *AU* individual es detectada y el sistema de clasificación facial la categoriza de acuerdo con la combinación de *AUs*. La Tabla 1.1 muestra las principales unidades de acción facial. Así, por ejemplo, para la emoción de “sorpresa”, las *AUs* presentes son: *AU1*, *AU2*, *AU5*, *AU25* y *AU26* como se observa en la Figura 1.1. (Ghimire et al, 2017) extrae características de apariencia de una región específica dividiendo la región de la cara completa en regiones locales de dominio específico. La importancia de las regiones locales es determinada usando un enfoque de búsqueda incremental, el cual resulta en una reducción de la dimensión de características y en una mejor precisión de reconocimiento.

Para características híbridas, algunos enfoques han combinado características geométricas y de apariencia para complementar las debilidades de los dos enfoques y proveer mejores resultados en ciertos casos. Por ejemplo, en (Benitez-Quiroz, Srinivasan, y Martinez, 2016) proponen un algoritmo que reconoce *AUs* y sus intensidades en grandes bases de datos de expresiones faciales. Así estas imágenes son automáticamente etiquetadas con *AUs*, intensidades de *AUs* y categorías de emociones. Para la identificación de *AUs* utilizan estadísticas de segundo orden de los puntos de referencia de la cara (como distancias y ángulos entre puntos de referencia) y filtros de Gabor. Para categorizar las emociones utilizan un clasificador basado en subclases, el Análisis discriminante de la subclase de kernel (KSDA, *Kernel Subclass Discriminant Analysis*). La contribución de su enfoque es que alcanzan una precisión de reconocimiento alta corriendo en tiempo real.

La revisión de la literatura tanto del marco teórico como del estado del arte en reconocimiento de expresiones faciales nos permitió definir estrategias para la extracción, selección y obtención de características e iniciar el trabajo de reconocimiento de emociones centradas en el aprendizaje a partir de video de imágenes visibles. La metodología será explicada en los siguientes apartados.

Tabla 1.1 Unidades de acción facial (*AUs*)

Número <i>AU</i>	Nombre de la acción	Número <i>AU</i>	Nombre de la acción
0	Rostro Neutral	24	Presionando los labios
1	Levantar las cejas internas	25	Separar los labios
2	Levantar las cejas externas (lado derecho)	26	Dejar caer la mandíbula
4	Bajar las cejas	27	Estiramiento de la boca abriéndola
5	Levantar parpado superior	28	Chuparse los labios
6	Levantar mejillas	29	Sacar mandíbula
7	Apretar parpado(s)	30	Mandíbula hacia los lados
8	Labios encimados uno de otro	31	Apretar mandíbula
9	Arrugar la nariz	32	Mordida labial
10	Levantamiento del labio superior	33	Succión de mejillas (soplar)
11	Profundidad naso labial	34	Inflar mejillas
12	Tiramiento labial esquinial	35	Chupar
13	Tiramiento labial frontal	36	Protuberancia de lengua (bulto)

14	Olluelos	37	Repasar los labios con la lengua
15	Bajar las esquinas de los labios	38	Dilatar fosas nasales
16	Bajar el labio inferior	39	Comprimir fosas nasales
17	Levantar la barbilla o mentón	41	Dejar caer el parpado superior (sin cerrar)
18	Arrugar los labios	42	Entre cerrar los ojos
19	Mostrar la lengua	43	Ojos cerrados
20	Ensanchar (estirando) los labios	44	Vistazo (ojos pequeños, como bizco)
21	Apretamiento del cuello	45	Parpadear
22	Poner en embudo los labios	46	Guiño
23	Tensor los labios (enfrente en pico)		



Figura 1.1 Unidades de acción facial presentes en la emoción "sorpresa" (AU1, AU2, AU5, AU26 y AU25).

### 3 Metodología propuesta

Las etapas involucradas en la metodología propuesta corresponden a la implementación del proceso para el descubrimiento de datos en bases de datos.

1. **Adquisición de datos** (*selection*): corresponde a la digitalización de los datos y su almacenamiento.

2. **Etiquetado de los datos** (*target data*): consiste en seleccionar e integrar el conjunto de los datos provenientes de fuentes múltiples y heterogéneas. En la primera etapa se etiqueta el video visible por observadores humanos y posteriormente se deben etiquetar fotogramas y videos de manera automática (considerando las etiquetas asignadas por el observador humano). Con los datos etiquetados de video visible se iniciará el preprocesamiento de datos y se analizará su posible integración. En una segunda etapa etiquetaremos considerando la autoevaluación que el estudiante realizó al contestar la prueba de emociones (lo que llamaremos auto etiquetamiento), etiquetando las variables continuas de las emociones. Se etiquetará cada variable (valencia, activación y control) con la emoción discreta seleccionada por el alumno en el instante de tiempo en el que contesta el test de emociones.

3. **Preprocesamiento** (*preprocessing*): si es necesario, eliminación de ruido y de datos aislados. Uso del conocimiento previo para eliminar inconsistencias y duplicados. Elección y uso de estrategias para manejar la información faltante de los conjuntos de datos. En video se debe mejorar el contraste de la imagen, usando equalización del

histograma. Posteriormente se detecta y segmenta de forma automática la cara, eliminando el fondo de acuerdo con (Morales-Vargas, Reyes-García y Peregrina-Barreto, 2017), bajo los siguientes pasos:

a) Detección de puntos de referencia de la cara: consiste en identificar puntos de referencia de ojos, cejas, boca, nariz y contorno de la cara. Para la identificación de puntos de referencia faciales se utilizará el algoritmo AAM (*Active Appearance Model*), que recuperan 68 pares de puntos faciales  $st = \{\{x_i, y_i, x_2, y_2, \dots, x_{68}, y_{68}\}, t\}$  los cuales describen la forma de la cara; cada par corresponde a un vértice de un descriptor de la cara definido para cada imagen  $I$  de la secuencia de imágenes  $is = \{I, t\}$  de un video que corresponde a una etiqueta específica (aburrido o interesado). Donde  $is$ , es la secuencia de imágenes,  $I$  es un fotograma específico en un momento de tiempo  $t$ .

b) Alineación de puntos de referencia facial: como los puntos de referencia facial pueden ser afectados por la orientación, localización y escala (tamaño) y esto a su vez reflejarse en la eficiencia de los algoritmos de reconocimiento de expresiones, se debe utilizar un método para alinear los puntos de referencia. Un modelo heurístico basado en transformaciones afines será utilizado. Las transformaciones consisten en:

- Rotación de puntos de referencia para alinear horizontalmente los cantos de los ojos ( $s$ ).
- Cálculo del tamaño máximo del estado neutral ( $sz$ ) de la cara para aplicar la normalización con base al tamaño máximo.
- Normalización de los puntos de referencia ( $s, sz$ ) entre  $[0, 1]$ .
- Superposición del estado neutral por translaciones afines: corrección, para mantener las deformaciones causadas por los movimientos faciales y translación, para alinear estado neutral con la expresión facial. El fotograma 1 de cada secuencia de video etiquetado será considerado como el estado neutral.

4. **Transformación de los datos** (*transformation*): aquí se lleva a cabo la selección de características útiles para representar los datos, la reducción de dimensionalidad o métodos de transformación y fusión de los datos. Se obtienen las características geométricas y *AUs* de acuerdo con (Morales-Vargas, Reyes-García y Peregrina-Barreto, 2019):

**Obtención de características geométricas:**

- 1) Cálculo de la magnitud y orientación de los puntos de referencia del fotograma inicial ( $o$ ) y del fotograma final ( $f$ ).

$$m_i = \sqrt{(dx_i)^2 + (dy_i)^2} \quad \text{y} \quad o_i = \tan^{-1} \left( \frac{dx_i}{dy_i} \right)$$

- 2) Concatenación de las tuplas (magnitud y orientación) por punto de referencia facial en un vector de la forma:  $mo_i = [m_1, o_1; m_2, o_2; \dots; m_n, o_n]$ .

- 3) Obtención de una forma triangular de los puntos de referencia facial:

$$ts = [a_1, b_1, c_1; a_2, b_2, c_2; \dots; a_n, b_n, c_n]$$

- 4) Cálculo de las áreas de cada triángulo en el vector:  $at = [a_1; a_2; \dots; a_j]$  con  $j$  siendo el número de triángulos (en total se calcula el área de 107 triángulos por fotograma).

- 5) Usando las formas triangulares se calculan los cambios en tamaño de las áreas faciales del fotograma actual ( $t=f$ ) respecto al fotograma neutral ( $t=0$ ).

$$a_i = at_{i,f} - at_{i,0}$$

6) Concatenación de los vectores de características  $mo$  y  $a$  en el vector  $br=[mo,a]$  de dimensión 243 (68X2=136 de  $mo$  y 107 de  $a$ ).

7) Reducción de la dimensionalidad por medio de una exploración manual de los datos considerando los movimientos en las unidades de acción que están presentes en las emociones de aburrido e interesado y en las emociones básicas. Para esto se seleccionan en cuanto a magnitud y orientación: cejas internas, cejas externas, parpados de los ojos, nariz, esquina derecha del labio, esquina izquierda del labio, labio superior, labio inferior y esquinas de los labios (10 partes faciales). En cuanto a las áreas de los triángulos se seleccionan ojos y boca. De esta forma se reduce la dimensionalidad original de 243 a 22 para el vector de características  $br$ .

8) Para incluir todas las características de una manera más simple se utilizarán dos operaciones de agrupación:

Agrupación Promedio 
$$PooledCar = \frac{1}{p} \sum_{i=1}^p v_i \quad y$$

Agrupación Máxima  $PooledCar = \max(v_i)$ , donde  $i$  es la  $i$ -ésima área distintiva a considerar y  $v_i$  es su subconjunto de puntos de referencia.

#### **Obtención del vector de características de AUs**

1)  $PooledCar$  es el vector de características con el que se entrena un modelo de inferencia fuzzy para cada unidad de acción.

2) En conjunto todo el sistema es usado para describir los movimientos faciales de cada secuencia en términos de AUs, obteniendo un vector de características  $\mu_{AU}$  con un valor de membresía para cada AU.

**5. Minería de datos (data mining):** consiste en elegir los algoritmos de aprendizaje computacional, la tarea de minería de datos y los algoritmos de minería que traten todos los criterios (clasificación, regresión, clusterización o modelos mixtos). Se iniciará con una SVM con los vectores de característica  $\mu_{AU}$  generados a partir del etiquetado manual por cada etiquetador humano y con los valores de las variables del modelo de emociones continuo (auto etiquetamiento). También se plantea entrenar una Red Neuronal de Aprendizaje Profundo (DNN, *Deep Neural Network*) a partir del video visible.

**6. Identificación de patrones (patterns):** buscar patrones de interés en una forma en particular, es decir, el reconocimiento de emociones.

**7. Interpretación y evaluación de los patrones minados (interpretation / evaluation):** se realiza la interpretación y evaluación de los resultados de los algoritmos utilizados.

**8. Consolidación del conocimiento descubierto (knowledge):** discusión y análisis de la relación de las emociones identificadas con el nivel de aprendizaje.

## 4 Implementación

1. Etiquetado de video visible: dos etiquetadores humanos analizaron 57 videos, la información registrada es almacenada en un archivo de Excel cuyo esquema se muestra en la Tabla 1.2, donde  $t_0$  es el tiempo en el que inicia la emoción identificada,  $t_f$ , el momento en el que culmina la emoción y *etiqueta* es el nombre de la emoción discreta identificada por observación de las AUs.

Tabla 1.2 Etiquetamiento manual de videos

<i>t<sub>0</sub></i>		<i>t<sub>f</sub></i>		<i>etiqueta</i>
min	seg	min	seg	Interesado/Aburrido
0	0	2	33	Interesado
2	33	2	35	Aburrido
2	35	17	10	Interesado
17	10	18	0	Aburrido
18	0	19	37	Interesado.....

2. Para poder llevar a cabo el etiquetamiento manual se identificaron las AUs presentes con mayor frecuencia en cada uno de los estados: interesado y aburrido (de acuerdo con el consenso de los etiquetadores humanos).

3. Auto etiquetamiento, se realizó considerando la etiqueta seleccionada por el alumno en el momento en el que contesta el test de emociones. Por cada video se selecciona el minuto completo en el que se encuentra a la mitad la hora registrada en la que el alumno contesta el test de emociones. Por lo tanto, para cada experimento obtenemos 3 videos de un minuto de duración cada uno etiquetado por la emoción discreta asignada por el propio alumno.

4. Para las dos propuestas de etiquetamiento, se hace la implementación automática de etiquetado. Con el video visible se generan videos para el rango de tiempo especificado para cada etiqueta y también fotogramas individuales correspondientes al mismo video. El video se genera a un rango de 1 fotograma por segundo con la idea de reducir el número de fotogramas a procesar y eliminar redundancias.

5. Mejorar el contraste de la imagen, usando ecualización del histograma. La finalidad es resaltar las diferentes partes de la cara para facilitar la identificación de los puntos de referencia faciales.

6. Detección y segmentación automática de la cara, eliminando el fondo. Consiste en la detección de puntos de referencia de la cara: la identificación de los puntos de referencia de ojos, cejas, boca, nariz y contorno de la cara se hace con el algoritmo de apariencia activa (AAM), que recupera 68 pares de puntos que describen la forma de la cara.

7. Alineación de puntos de referencia facial: como los puntos de referencia facial pueden ser afectados por la orientación, localización y escala (tamaño) y esto a su vez reflejarse en la eficiencia de los algoritmos de reconocimiento de expresiones, se utiliza un método para alinear los puntos de referencia a través de un modelo heurístico basado en transformaciones afines (rotación, normalización, traslación y corrección).



8. Obtención del vector de características geométricas: cálculo de la magnitud y orientación de los puntos de referencia del fotograma neutral ( $t=0$ ) y del fotograma final ( $t=f$ ). Se genera el vector  $mo$ .

9. Obtención de una forma triangular de los puntos de referencia facial. De acuerdo con el esquema planteado en la metodología, se genera el vector de áreas ( $at$ ) resultado de triangular la cara a partir de los puntos de referencia facial.

10. Usando las formas triangulares se calculan los cambios en tamaño de las áreas faciales del fotograma actual ( $t=f$ ) respecto al fotograma neutral ( $t=0$ ). Se genera el vector  $a$ . Se concatenan los vectores de características  $mo$  y  $a$  obteniendo el vector:  $br = [mo, a]$  de dimensión 243. Se reduce la dimensionalidad del vector  $br$  de 243 a 22 para el vector de características  $br$ .

11. Para incluir todas las características de una manera más simple se realiza una agrupación bajo dos esquemas diferentes: promedio y máximo generando el vector de características, *PooledCar*.

12. Obtención del vector de características de *AUs*. El vector de características *PooledCar* (promedio o máximo), fue utilizado para entrenar un modelo fuzzy para el reconocimiento de las *AUs*. El modelo fuzzy genera como salida un vector de características  $\mu_{AU}$  con un valor de membresía para cada *AU*. En la Tabla 1.3, se pueden observar los vectores  $\mu_{AU}$  para los primeros 10 fotogramas de uno de los videos procesados. Los valores de las columnas corresponden al valor de membresía de cada *AU* (considerando las primeras 27 *UAs*). El primer vector corresponde al estado neutral  $t(0)$ . Un valor de cero indica que esa *AU* no está presente. Un valor negativo representa un movimiento contrario al de esa *AU*. Estos vectores de características son los que se utilizaran para entrenar algoritmos de clasificación, iniciando con una máquina de vectores de soporte.

Tabla 1.3 Ejemplo de vectores de características  $\mu_{AU}$ .

# FRAME	UNIDADES DE ACCIÓN FACIAL																										
	1	2	3	4	5	6	7	23	24	25	26	27															
1	0	0	0	0	0	0	0	0	0	0	0	0															
2	0.086920353	-0.087781499	0	0.191393858	-0.115772166	0	0.13792959	0.095066791	-0.086976008	-0.016134692	-0.009234107	-0.095820488															
3	0.400204895	0.469842737	0	0.326715145	-0.064400575	0	0.142108596	0.002124084	-0.090291585	0.027900686	-0.0184323	-0.056019226															
4	0.383735395	0.511373163	0	0.33365076	-0.004739833	0	0.196583807	0.015750755	0.294621725	0.037929306	-0.019396031	-0.053118764															
5	0.363436106	1.034559102	0	0.454895283	-0.040631021	0	0.046459533	0.0404546	-0.166587478	0.023634897	-0.02152879	-0.043505756															
6	0.516778277	1.02091871	0	0.448931474	0.023652363	0	0.079359396	0.319357178	0.165060629	-0.031243383	-0.02580574	-0.026299897															
7	0.694182616	1.117885516	0	0.498037026	0.10990929	0	-0.021940848	0.162751288	0.140385646	-0.029428976	-0.025450557	-0.027897326															
8	0.748497556	0.649206871	0	0.357454036	0.014655282	0	0.133615624	0.282843494	-0.44856777	0.001421282	-0.022706122	-0.038698736															
9	0.748497556	0.649206871	0	0.357454036	0.014655282	0	0.133615624	0.282843494	-0.44856777	0.001421282	-0.022706122	-0.038698736															
10	0.493736697	1.043212336	0	0.457062675	0.121652715	0	-0.046137855	0.140860713	-0.850049925	-0.043865249	-0.019294491	-0.052444062															

## 5 Conclusiones

El etiquetamiento de video es la actividad más importante para una buena clasificación. Hacerlo de manera automática bajo las estrategias de etiquetado humano y auto etiquetado reduce el tamaño del video y nos permite obtener características más precisas para cada emoción. Después de obtener las características de cada fotograma, la selección de las

más representativas para cada una de las dos emociones representó una reducción en la dimensionalidad lo que a su vez disminuye la complejidad del procesamiento de los datos. Finalmente, los vectores de cada *AUs* (cuyos valores indican la presencia de cada *AUs* en la cara de los individuos) serán la entrada a algoritmos de aprendizaje máquina. Este es el siguiente paso en la clasificación de emociones centradas en el aprendizaje (interés y aburrimiento).

## Referencias

- Benitez-Quiroz, C. F., Srinivasan, R., y Martinez, A. M. (2016). "EmotioNet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild". *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December, pp. 5562–5570.
- Ekman, P., Friesen, W., y Hager, J. (2002). "Facial Action Coding System. The Manual", Salt Lake City, United States of America.
- Fayyad, U., Piatetsky, G., y Smyth, P. (1996). "From Data Mining to Knowledge Discovery in Databases", *AI Magazine*, vol. 17, pp. 37–54.
- Ghimire, D., Jeong, S., Lee, J., y Park, S. H. (2017). "Facial expression recognition based on local region specific features and support vector machines". *Multimedia Tools and Applications*, vol. 76(6), pp. 7803–7821.
- Ghimire, D., & Lee, J. (2013). "Geometric feature-based facial expression recognition in image sequences using multi-class AdaBoost and support vector machines". *Sensors (Switzerland)*, vol. 13(6), pp.7714–7734.
- Hjortsj, C.-H., Ekman, P., Friesen, W. V, Hager, J. C., FACS, F., y FACS, F. M. (2019). "Sistema de Codificación Facial", pp. 1–8.
- Morales-Vargas, E., Reyes-García, C. A. y Peregrina-Barreto, H. (2017). "Reconocimiento de expresiones faciales con base en la dinámica de puntos de referencia faciales". *In Research in Computing Science*, vol. 140, pp. 9–18.
- Morales-Vargas, E., Reyes-García, C. A. y Peregrina-Barreto, H. (2019). "On the use of action units and fuzzy explanatory models for facial expression recognition". *PLoS ONE*, vol. 14(10), pp. 1–13.
- Walecki, R., Rudovic, O., Pavlovic, V., Schuller, B., y Pantic, M. (2017). "Deep structured learning for facial action unit intensity estimation", *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5709–5718.

# Capítulo 4

## Avances en la detección y predicción de trayectorias de obstáculos en ambientes vehiculares mediante visión computacional

Lauro Reyes-Cocoletzi, Ivan Olmos-Pineda, Arturo Olvera-López

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

lrc07reyes@mail.com, {iolmos, aolvera}@cs.buap.mx

**Resumen.** En este artículo se muestra el panorama general de las actividades realizadas para la predicción de trayectorias en tráfico vehicular por medio de visión por computadora. Se realizan pruebas iniciales en cuanto a los distintos bloques que conforman el algoritmo propuesto en este trabajo de investigación. Por ejemplo, para el bloque encargado de la detección de objetos por medio de video se realizan pruebas en escenas vehiculares bajo condiciones ambientales de luz en exteriores, así como con archivos de video de tránsito disponibles en bases de datos de uso libre. Para el bloque de seguimiento de trayectoria se realiza una aproximación en el rastreo de objetos en escena y se estima la localización espacial de los objetos de interés conforme se desplazan. Finalmente se une lo realizado en cada bloque para presentar experimentos y resultados iniciales del funcionamiento completo del algoritmo propuesto.

**Palabras Clave:** detección de objetos, estimación de trayectoria, visión computacional.

### 1 Introducción

Actualmente los esfuerzos realizados en la industria automotriz han incrementado para alcanzar vehículos autónomos que se desplacen sin riesgo de colisión. Los vehículos disponibles en el mercado cuentan con piloto automático de carácter funcional sólo para autopistas con condiciones de tráfico específico; en algunos otros casos se tiene un equipo asistente para rutas en tráfico denso que funciona únicamente en circunstancias específicas (normalmente sólo por debajo de velocidades de 60 o 65 Km/h).

Los organismos de regulación de transporte (por ejemplo, National Highway Traffic Safety Administration NHTSA) establecen la necesidad de contar con un sistema capaz de solventar circunstancias imprevistas en el desplazamiento autónomo, por ejemplo, si se produce un error o las condiciones óptimas de funcionamiento desaparecen, debe existir un

sistema de respaldo capaz de tomar el control y sostener la conducción autónoma hasta recuperar las condiciones de riesgo mínimo (Mukar et al, 2015).

Debido a la necesidad de sistemas robustos de conducción, los trabajos de investigación en la industria automotriz, así como en centros de investigación continúan para alcanzar el nivel de autonomía completo.

Algunas de las limitaciones en el procesamiento de la información para llevar a cabo el desplazamiento autónomo son: bajo rendimiento en función de la tasa de detección, número limitado de objetos percibidos como obstáculos (generalmente sólo se toman en consideración la detección de vehículos y peatones), información limitada del ambiente vía los sensores utilizados, alto costo de los sensores (radar laser) y alto tiempo de procesamiento, entre otras problemáticas (Roggeman et al, 2017).

En este trabajo de investigación se propone la adquisición de información del ambiente mediante un sistema de cámaras estéreo debido a su bajo costo. Además, el contar con dos cámaras es útil ya que da una aproximación de profundidad del ambiente y puede obtenerse una estimación tridimensional de la información capturada en 2D de los objetos presentes en escena.

Los avances en el algoritmo propuesto comprenden cinco etapas las cuales son: pre-procesamiento, extracción de características, detección, seguimiento y estimación de posición. La etapa de pre-procesamiento abarca la tarea de calibración de las cámaras de video para obtener los parámetros de ajuste de los píxeles en dos imágenes estéreo. A partir de las imágenes estéreo acopladas se puede tener información espacial de los objetos en escena a través de la obtención de mapas de disparidad.

La etapa de extracción de características implica resaltar la información relevante en regiones del cuadro de video en las cuales se puede decir que existen objetos. Las características utilizadas para establecer que hay un objeto en escena constan del umbral de color junto con la información de posición espacial de las regiones de interés.

La detección del objeto implica establecer las zonas o regiones de interés en el cuadro de video que contienen obstáculos. Esta etapa está estrechamente relacionada con la etapa de extracción de características por lo que también se hace uso de una red neuronal convolucional para actualizar y validar la presencia de un objeto en escena.

Las etapas de seguimiento y estimación de posición consisten en rastrear las regiones de interés detectadas en las etapas anteriores y prever el desplazamiento de los objetos durante los siguientes cuadros del video. El seguimiento se realiza por medio de los algoritmos de filtros de correlación (Kernelized Correlation Filters), la estimación de trayectoria se procesa por el momento a través de filtros de Kalman.

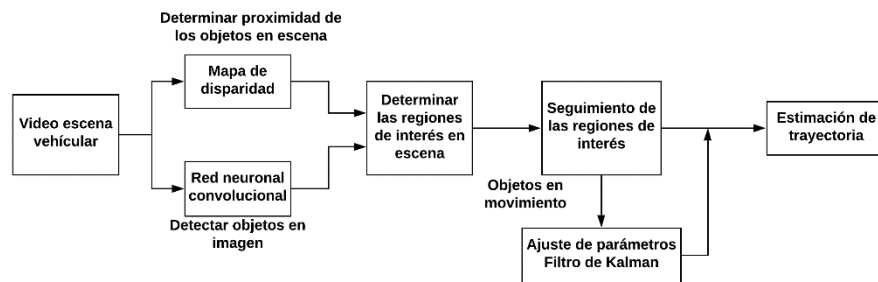
En este trabajo se presentan los experimentos realizados de cada etapa del algoritmo de estimación de trayectorias con videos tomados de ambientes reales y videos adquiridos de bases de datos libres.

El trabajo realizado se organiza en las siguientes secciones: en la sección correspondiente a la metodología se describe el procedimiento que se está efectuando para alcanzar los objetivos. En la sección de parámetros de evaluación se presentan las métricas para la evaluación del seguimiento. Se muestra en la sección de experimentos los resultados preliminares obtenidos. Los datos e información recopilada de las pruebas hechas sirven

para establecer conclusiones parciales y el trabajo a futuro que se debe realizar para alcanzar los objetivos establecidos.

## 2 Metodología

El algoritmo inicialmente propuesto está estructurado en etapas específicas. La figura 1 muestra el conjunto de etapas y su descripción se menciona en los párrafos siguientes.



**Figura 1.** Bloques para la estimación de trayectorias de objetos en escena vehicular.

La parte inicial es la percepción del entorno mediante un par de cámaras (visión estereoscópica) junto con el pre-procesamiento. La parte fundamental de la etapa de detección es realizar un mapeo de disparidad para simplificar la cantidad de información a procesar en la escena. El mapeo de disparidades obtiene características básicas para la representación de objetos en una escena en entornos de tránsito complejos, además tiene la ventaja de proporcionar información espacial de la escena (Gigaris y Komodakis, 2015).

Las siguientes etapas incluyen el procesamiento basado en una red neuronal convolucional (RNC), por medio de esta red es posible realizar la detección de los obstáculos que se encuentran en la escena.

La red neuronal recibe una imagen de entrada (cuadro del video), divide la imagen en una cuadrícula de celdas de 13 x 13 píxeles. El tamaño de estas 169 celdas varía según el tamaño de la entrada. Por ejemplo para un tamaño de imagen de entrada de 416 x 416 píxeles, el tamaño de la celda es de 32 x 32 píxeles. Cada celda es responsable de predecir una serie de cuadros en la imagen (Redmon et al, 2016).

Para cada cuadro delimitador, la red también predice la confianza de que el cuadro delimitador realmente encierra un objeto y la probabilidad de que el objeto encerrado sea una clase particular. La mayoría de estos cuadros delimitadores se eliminan porque su confianza es baja o porque encierran el mismo objeto que otro cuadro delimitador con un puntaje de confianza muy alto, esta técnica se llama supresión no máxima.

Al comparar la región donde se detectó objeto dada por la RCN con el mapa de disparidad construido se valida la presencia de obstáculos con riesgo de colisión en el cuadro del video.

Los obstáculos detectados deben seguirse en los cuadros subsiguientes del video mientras presenten un riesgo de colisión. Se propone que el bloque de seguimiento se resuelva identificando una región conectada en la imagen (estas regiones en la literatura se conocen como blob) asociado con cada vehículo y posteriormente rastrear a lo largo del tiempo utilizando una medida de correlación cruzada.

En este caso, la estimación de la posición actual (en la imagen) y la estimación de la trayectoria están conectadas, esto se puede denotar como algoritmos cuya tarea es el seguimiento de múltiples objetos. La propuesta de este trabajo plantea resolver el problema de estimación de trayectoria a través del seguimiento por detección para calcular la estimación de la siguiente posición de los obstáculos en el tráfico vehicular.

Finalmente, la estimación de la trayectoria se trata como un problema de probabilidad, las trayectorias del objeto se pueden considerar como un grupo de puntos que se mueven (en el espacio y el tiempo) simultáneamente. Sobre la base de la posición de la información anterior y la ruta seguida por la etapa de seguimiento, se establece una hipótesis de trayectoria (Wheeler et al, 2016).

Se establecen parámetros para la hipótesis de la trayectoria para modelar matemáticamente la estimación de movimiento basado en la distribución de probabilidad estadística (aún pendiente por realizar).

### 3 Parámetros de evaluación

En el pasado, se ha propuesto una gran cantidad de métricas para la evaluación cuantitativa del seguimiento de múltiples objetivos. Elegir el parámetro de evaluación depende en gran medida de la aplicación.

Por un lado, es preferible resumir el rendimiento en un solo número para permitir una comparación directa. Por otro lado, no se debe perder información sobre los errores individuales cometidos por los algoritmos y poder proporcionar la estimación general del rendimiento (Alahi et al, 2016).

Existen dos métricas utilizadas frecuentemente en la literatura para la evaluación del seguimiento de objetos en movimiento las cuales son MOTA (Multiple Object Tracking Accuracy) y MOTP (Multiple Object Tracking Precision).

MOTA es quizás la métrica más utilizada para evaluar el rendimiento de un rastreador. La razón principal de esto es que combina tres fuentes de errores definidos (Barth et al, 2010). Los errores mencionados son el falso positivo *FP*, falso negativo *FN* y la equivalencia en cambio de identidad IDWS (equivalently an identity switch).

El parámetro MOTA puede ser negativo en casos donde el número de errores cometidos por el rastreador excede el número de todos los objetos en la escena GT (ground truth

objects); cabe señalar que  $t$  se considera como el índice del número consecutivo de cuadros a analizar en el video.

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \quad (1)$$

Aunque el puntaje MOTA da una buena indicación del rendimiento general, es muy discutible si sólo este número puede servir como única medida de rendimiento; por lo que otro parámetro de evaluación usado de forma recurrente es MOTP.

MOTP es la diferencia promedio entre todos los verdaderos positivos y sus correspondientes objetivos GT (Barth et al, 2010). Para la superposición del cuadro delimitador, esto se calcula como:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \quad (2)$$

donde  $c_t$  denota el número de coincidencias en el cuadro  $t$  y  $d_{t,i}$  es la superposición del cuadro delimitador del objetivo  $i$  con el objeto GT asignado (Barth et al, 2010).

MOTP da la superposición promedio entre todas las hipótesis correctamente coincidentes y sus respectivos objetos con rangos entre 50 % y 100 %. Es importante señalar que MOTP es una medida de precisión de localización, que no debe confundirse con el valor predictivo positivo o la relevancia en el contexto de las curvas de precisión/recuerdo utilizadas en la detección de objetos (Lee et al, 2016).

## 4 Experimentos

Para este trabajo se realizaron experimentos para analizar los resultados en la etapa de detección de objetos (algoritmo mapas de disparidad) así como la etapa de seguimiento de objetos (Kernelized Correlation Filters).

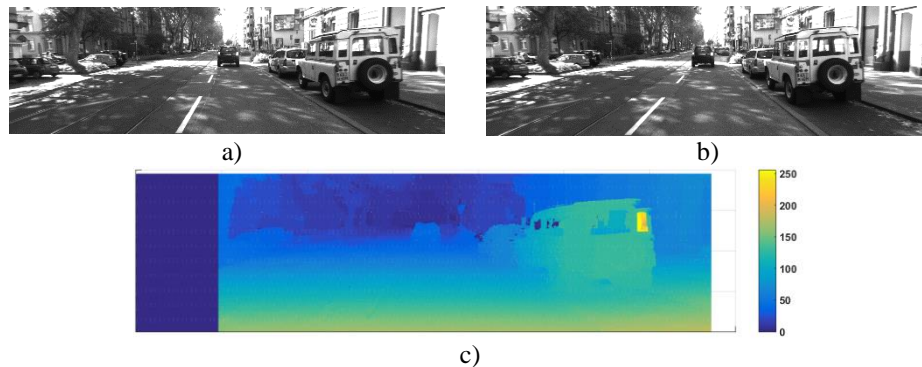
A continuación se muestran el procesamiento de videos capturados de ambientes de tránsito real y de videos en condiciones controladas (repositorios en base de datos) para evaluar la detección y seguimiento de los objetos en escena.

### 4.1 Procesamiento en el video

El algoritmo de mapas de disparidad es una primera aproximación para detectar los objetos existentes en la escena; sin embargo, es necesario detallar y corroborar la presencia de un objeto con fundamento en los datos obtenidos al emplear este método.

En la figura 2 se muestra una escena vehicular controlada (video adquirido de la base de datos KITTI Vision Benchmark Suite Ristani et al, 2016) y el mapa de disparidad procesado. Cualitativamente se puede observar que las zonas más claras señalan que

objetos se encuentra más cerca del dispositivo de captura (cámaras estéreo), por lo tanto las zonas oscuras denotan los objetos más alejados.

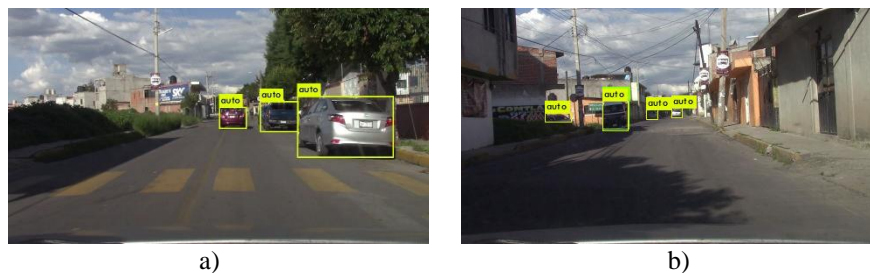


**Figura 2.** a) Imagen izquierda capturada, b) imagen derecha capturada, c) mapa de disparidad obtenido al procesar el par de imágenes.

Cuantitativamente los valores obtenidos para cada pixel al procesar el algoritmo de disparidad se encuentran en un rango de 0 a 255 unidades. Los valores entre 255 y 200 indican posición espacial de los objetos cercanos a las cámaras y valores entre 50 y 0 indican posición espacial de los objetos en el fondo de la escena.

Aún se deben eliminar píxeles aislados y bordes de escasa longitud para evitar falsos positivos o falsos negativos al establecer las regiones donde se detecta un objeto. También se necesita delimitar las regiones a seguir, dentro de las cuales se encuentra el objeto, esto implica la detección de características específicas para referencia y seguimiento de las mismas a través del tiempo.

La RCN se encarga de resolver las necesidades antes mencionadas, en la figura 3 se muestra el desempeño de la red utilizada (DarkNet-yolo) en imágenes capturados en calles de Puebla, México.



**Figura 3.** a) Objetos detectados relativamente cerca de las cámaras, b) objetos relativamente lejos de las cámaras.



Las coordenadas espaciales obtenidas por la red neuronal se comparan con las regiones entre los valores 200 - 255 en los mapas de disparidad para validar la presencia de objetos y además determinar cuáles de ellos representan un riesgo de colisión (debido a la cercanía espacial).

El siguiente experimento consiste en rastrear los grupos de píxeles que se mueven juntos (objetos) y trazar puntos de referencias espaciales a partir de las posiciones previas para el seguimiento del desplazamiento.

La figura 4 muestra una escena de tráfico capturada en el entorno real de una calle típica de Puebla, México donde se puede observar vehículos en movimiento. Cada objeto de interés está delimitado por un cuadro envolvente que denota un umbral de valores representativos del objeto.



**Figura 4.** Seguimiento de objetos en escena vehicular; a) cuadros envolventes de las regiones con las características de interés en tiempo  $t$ , b) seguimiento en el cuadro consecutivo, tiempo  $t+1$ .

Los píxeles dentro del cuadro envolvente delimitador están etiquetados por un vector de movimiento, que indica el cambio en los datos de la imagen del tiempo  $t$  al tiempo  $t + 1$ , de tal manera que sigue la trayectoria tomada por los píxeles de interés detectados.

La información de desplazamiento de píxeles analizada en este experimento proporciona solo una aproximación preliminar de las regiones locales en la imagen donde se deben detectar los obstáculos.

Finalmente en lo que respecta al bloque de estimación de trayectoria únicamente se realizaron experimentos sencillos utilizando el algoritmo de Filtro de Kalman en donde se estima la posición siguiente de un grupo de píxeles con características comunes de color (figura 5).



**Figura 5.** El círculo rojo denota la región de interés y el círculo amarillo la estimación de posición en el a) cuadro inicial y en el b) cuadro consecutivo.

El resultado obtenido al aplicar el algoritmo de Kalman a un video (tomado de una base de datos) muestra la posición actual de las regiones de interés y la posición estimada siguiente. Cabe señalar que la estimación procesa la posición espacial siguiente en un rango de 100 a 200 pixeles de distancia de la posición actual.

Este rango es relativamente pequeño por lo que no proporciona la suficiente información como para definir si el móvil cambiará de dirección a la izquierda, derecha o seguirá de frente. La tarea por realizar implica mejorar el bloque de predicción de trayectoria, por lo que se plantea sustituir el algoritmo de Kalman utilizado en este experimento por un modelo causal.

Los resultados de los experimentos presentados hasta el momento son del tipo cualitativo, sin embargo para dar mayor peso a la parte experimental se procede a realizar la evaluación cuantitativa. A continuación se presenta la evaluación y resultados de las métricas establecidas en la sección 3.

El marco de referencia en los trabajos relacionados establecen valores objetivo de MOTA entre  $54.8 \pm 19.3$  % en la tasa de rendimiento, para MOTP el valor deseado se encuentra entre 52% y 65 %.

En la tabla 1 se muestran la evaluación de los resultados experimentales realizados con respecto al seguimiento de objetos. Se procesan un conjunto de videos capturados en escenarios reales y se obtienen los parámetros para evaluar las métricas establecidas.

**Tabla 1.** Resultados obtenidos al evaluar el seguimiento de objetos en escena.

<i>Evaluación del seguimiento (parámetros)</i>						
<i>Video 30 (fps)</i>	<i>Resolución</i>	<i>Long. (seg)</i>	<i>MOTA (%)</i>	<i>MOTP (%)</i>	<i>FP</i>	<i>FN</i>
1	960x540	14	33.7	56.5	96	68
2	960x540	10	33.2	50.2	89	102
3	960x540	20	46.6	56.4	81	92
4	960x540	35	26.2	50.5	95	120
5	960x540	20	34.6	53.2	63	51

El rango MOTA obtenido se encuentra entre 33.2 % a 46.6 %, para el caso de MOTP entre 50.2 % y 56.5 %, para los FN y FP los valores oscilan entre 51 y 120 objetos detectados.

Basado en los resultados obtenidos en los parámetros de MOTA y MOTP aún se debe trabajar en cada etapa del algoritmo de predicción de trayectorias para alcanzar los resultados reportados en los trabajos relacionados.

## 5 Conclusiones

Este documento describe brevemente el trabajo relacionado con la detección de obstáculos en entornos reales, presenta una descripción general, analiza las técnicas existentes y cómo se ha propuesto resolver el problema.

A partir de los experimentos realizados, se puede concluir que el problema de la detección de obstáculos es un área de interés y aún hay margen de mejora así como oportunidades para la innovación. Existe investigación por realizar, de manera que vale la pena continuar con el desarrollo de técnicas y herramientas para dar solución a la problemática planteada.

Cabe mencionar que debido a que este artículo solo presenta un avance y no el trabajo de investigación concluido es de suponerse que los resultados obtenidos presenten áreas de oportunidad.

## 6 Trabajo a futuro

El trabajo por realizar consta de llevar a cabo la integración completa de los bloques de la red neuronal convolucional junto con el mapa de disparidad para incrementar la tasa de detección, mejorar los resultados en cuanto a las métricas de evaluación y desarrollar la estructura del modelo de probabilidad a realizar para la estimación de trayectorias.

El modelo propuesto para estimar trayectorias a través de métodos de probabilidad se debe enfocar en determinar el desplazamiento futuro de los obstáculos detectados dentro de un rango de confianza comparativo y equiparable a trabajos recientes.

## Referencias

- Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., y Savarese, S. (2016). "Human trajectory prediction in crowded spaces", *2016 IEEE conference on computer vision and pattern recognition*, pp. 961–971.
- Barth, A., Siegemund, J., Meibner, A., Franke, U., y Forstner, W. (2010). "Probabilistic multi-class scene flow segmentation for traffic scenes", *2010 Joint Pattern Recognition Symposium*, pp. 503–512.
- Gidaris, S., y Komodakis, N. (2015). "Object detection via a multi-region and semantic segmentation-aware cnn model", *2015 Proceedings of the IEEE international conference on computer vision*, pp. 1134–1142.
- Lee, B., Erdenee, E., Jin, S., Nam, M.Y., Jung, Y.G., Rhee, P. K. (2016). "Multi-class multi-object tracking using changing point detection", *2016 European Conference on Computer Vision*, pp. 68–83.
- Mukhtar, A., Xia, L., Tang, B. (2017). "Vehicle detection techniques for collision avoidance systems", *2017 IEEE Transactions on Intelligent Transportation Systems*, pp. 2318–2338.

- Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). “You only look once: Unified, real-time object detection”, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788.
- Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C. (2016). “Performance measures and a data set for multi-target, multi-camera tracking”, *Proceedings of the European Conference on Computer Vision*, pp. 17–35.
- Roggeman, H., Marzat, J., Derome, M., Sanfourche, M., Eudes, A., Le Besnerais, G. (2017). “Detection, estimation and avoidance of mobile objects using stereo-vision and model predictive control”, *2017 IEEE International Conference on Computer Vision*, pp. 2090–2099.
- Wheeler, T. A., Robbel, P., y Kochenderfer, M. J. (2016). “Analysis of microscopic behavior models for probabilistic modeling of driver behavior”, *2016 IEEE 19th international conference on intelligent transportation systems (itsc)*, pp. 1604–1609.

# Capítulo 5

## Grafos embebidos y el cálculo eficiente de su Número de Conjuntos Independientes

Miguel Rodríguez-Hernández, Guillermo De Ita-Luna  
Doctorado en Ingeniería del Lenguaje y del Conocimiento. Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla,  
mikelrodriguez@gmail.com, deita@cs.buap.mx

**Resumen.** El problema de contar el número de conjuntos independiente de un grafo  $G$  (denotado como  $i(G)$ ) es un problema #P-Completo clásico. Presentamos algunos patrones en grafos que nos permiten el cálculo eficiente de  $i(G)$ . Por ejemplo, mostramos que para un grafo  $G$  donde su conjunto de ciclos puede ser arreglado como ciclos embebidos,  $i(G)$  se puede calcular eficientemente. Particularmente, nuestra propuesta cuenta conjuntos independientes en grafos outerplanar.

**Palabras Clave:** Reconocimiento de Patrones de Grafos, Conteo del número de Conjuntos Independientes, Conteo Exacto.

### 1 Introducción

Los problemas de conteo no solo son interesantes matemáticamente, sino que también surgen en muchas aplicaciones. Por ejemplo, si queremos saber la probabilidad de que un grafo permanezca conectado dada una probabilidad de falla de una arista, tenemos que aplicar algoritmos de conteo para aproximar tales probabilidades.

Considerando los problemas de conteo, el cálculo del número de conjuntos independientes de un grafo ha sido clave para determinar la frontera entre conteo eficiente y procedimientos de conteo intratables. Vadhan (2001) mostró que contar el número de conjuntos independientes en grafos de grado máximo 4 es #P – Completo. Greenhill (2000) mejoró el resultado previo mostrando que contar el número de conjuntos independientes en grafos de grado 3 es #P-Completo.

Siguiendo la línea de algoritmos exactos, Dahllöf y Jonsson (2002) han diseñado un método para contar conjuntos independientes y cuyo algoritmo exacto tiene como peor caso un límite superior de  $O(1.3247^n)$ , siendo  $n$  el número de vértices del grafo de entrada. Por otro lado, se han diseñado algoritmos eficientes para contar conjuntos independientes después de capturar relaciones estructurales en la topología de grafos, permitiendo diseñar patrones matemáticos especiales para contar conjuntos independientes solamente en ciertas topologías. Por ejemplo, Okamoto et al. (2005) han

propuesto un algoritmo lineal para contar el número de conjuntos independientes sobre grafos chordales.

Muchos problemas combinatorios cuestionan sobre el hecho de empotrar grafos dentro de otros objetos, Johnson (1982). Por ejemplo, el problema de planaridad de grafos pregunta si un grafo dado  $G$  puede ser embebido en el plano de tal forma que no se intersecten aristas (excepto en un punto extremo común). Reconocer si un grafo es planar o no puede realizarse en tiempo polinomial.

En nuestro caso, estamos interesados en una subclase particular de grafos planares, aquellos grafos cuyo conjunto de vértices puede ser arreglado como incidente con la cara externa (outerface), esta clase de grafos reciben el nombre de grafos outerplanar. Presentamos aquí, un algoritmo para contar el número de conjuntos independientes sobre grafos outerplanar.

## 2 Preliminares

Sea  $G = (V, E)$  un grafo no dirigido con conjunto de vértices  $V$  y conjunto de aristas  $E$ . Dos vértices  $v$  y  $w$  se dice que son adyacentes si existe una arista  $\{v, w\}$  que los conecta. Algunas veces, la notación de  $uv$  se usa para denotar la arista  $\{u, v\} \in E$ .

El *vecindario* para  $x \in V$  es  $N(x) = \{y \in V: \{x, y\} \in E\}$  y su *vecindario cerrado* es  $N(x) \cup \{x\}$  el cual se denota por  $N[x]$ . Denotamos la cardinalidad de un conjunto  $A$ , por  $|A|$ . El grado de un vértice  $x$ , se denota por  $\delta(x)$ , es  $|N(x)|$ . El grado de  $G$  es  $\Delta(G) = \max\{\delta(x): x \in V\}$ . El tamaño del vecindario de  $x$ ,  $\delta(N(x))$ , es  $\delta(N(x)) = \sum_{y \in N(x)} \delta(y)$ . Un vértice  $v$  es *colgante* si  $\delta(x)=1$ ; y una arista  $e = \{x, y\}$  es *colgante* si  $x$  o  $y$  es un vértice colgante.

Un camino de  $v$  a  $w$  es una secuencia de aristas:  $v_0v_1, v_1v_2, \dots, v_{n-1}v_n$  tal que  $v = v_0$  y  $v_n = w$  y  $v_k$  es adyacente a  $v_{k+1}$ , para  $0 \leq k < n$ . La longitud del camino es  $n$ . Un camino simple es un camino donde  $v_0, v_1, \dots, v_{n-1}, v_n$  son todos diferentes. Un ciclo es un camino no vacío tal que el primero y el último vértice son idénticos, y un ciclo simple es un ciclo en el cual no se repiten los vértices, excepto que el primero y el último vértice son idénticos. Un grafo  $G$  es acíclico si no tiene ciclos.  $P_n, C_n, R_n, K_n, N_n$  denotan respectivamente, un camino, un ciclo simple, una estrella con un nodo centro, el grafo completo y el conjunto de  $n$  vértices sin ninguna arista, respectivamente.

Dado un grafo  $G = (V, E)$ ,  $G' = (V', E')$  es un subgrafo de  $G$ , si  $V' \subseteq V$  y  $E'$  contiene aristas  $v, w \in E$  tal que  $v \in V'$  y  $w \in V'$ . Si  $E'$  contiene cada arista  $v, w \in E$  donde  $v \in V'$  y  $w \in V'$  entonces  $G'$  recibe el nombre de grafo inducido de  $G$ . Una componente conectada de  $G$  es un subgrafo inducido maximal de  $G$ , es decir, una componente conectada no es un subgrafo propio de ningún otro subgrafo conectada de  $G$ . Note que en una componente conectada, para cada par de sus vértices  $x, y$ , existe un camino de  $x$  a  $y$ . Si un grafo acíclico es también conectado, entonces éste es llamado un árbol libre.

Dado un grafo  $G = (V, E)$ ,  $S \subseteq V$  es un conjunto independiente en  $G$  si para cada par de vértices  $v_1, v_2$  en  $S$ ,  $\{v_1, v_2\} \notin E$ . Sea  $I(G)$  la notación para el conjunto de todos los

conjuntos independientes de  $G$ . Un conjunto independiente  $S \in I(G)$  es maximal si éste no es un subconjunto de ningún otro conjunto independiente de  $G$ , y es máximo si tiene el tamaño más grande entre todos los conjuntos independientes en  $I(G)$ . La determinación del conjunto independiente máximo ha recibido mucha atención ya que es un problema NP – completo.

El correspondiente problema de conteo sobre conjuntos independientes es denotado por  $i(G)$ , y éste consiste en contar el número de conjuntos independientes de un grafo  $G$ .  $i(G)$  es un problema #P-completo para grafos  $G$  donde  $\Delta(G) \geq 3$ . Calcular  $i(G)$  permanece #P-completo cuando se restringe a grafos 3-regulares, Greenhill (2000).

Existen diferentes procedimientos en tiempo polinomial para calcular  $i(G)$  cuando  $\Delta(G) \leq 2$  como se muestra en Dahllöf y Jonsson (2002), Roth (1996) y Russ (2001). En realidad, todos tienen complejidad en tiempo lineal. En las siguientes secciones presentamos procedimientos combinatorios exactos para calcular  $i(G)$  de acuerdo a patrones especiales que existen en los grafos.

### 3 Patrones básicos de grafos para el conteo eficiente de Conjuntos Independientes

Se sabe que el conteo de conjuntos independientes de un grafo  $G$ , se traslada al producto del número de conjuntos independientes de sus componentes conectadas, esto es,  $i(G) = \prod_{i=1}^k i(G_i)$  donde  $G_i, i=1, \dots, k$  son las componentes conectadas de  $G$  como se describe en Roth (1996). Además, la complejidad total en tiempo para calcular  $i(G)$ , denotado como  $T(i(G))$ , se da por la regla máxima como  $T(i(G)) = \max\{T(i(G)): G_i \text{ es una componente conectada de } G\}$ . De este modo, una primera descomposición útil del grafo se realiza mediante sus componentes conectadas. Por eso es que consideramos como un grafo de entrada solo una componente conectada. Iniciamos el análisis de los casos más simples de una componente conectada de un grafo.

#### Caso A:

Sea  $P_n = G = (V, E)$  un grafo que consiste de una sola secuencia de nodos (camino), es decir,  $V = \{1, 2, \dots, n\}$  y existe una arista  $e_i = \{i, i+1\}$ ,  $i = 1, \dots, n-1$ , para cada par de vértices consecutivos.

Construimos la familia  $f_i = \{G_i\}$ ,  $i=1, \dots, n$  donde cada  $G_i = (V_i, E_i)$  es el grafo inducido de  $G$  formado por los primeros  $i$  vértices de  $V$ .

Asociamos a cada vértice  $v_i \in V$  un par  $(\alpha_i, \beta_i)$  donde  $\alpha_i$  expresa el número de conjuntos en  $I(G_i)$  donde el vértice  $v_i$  no aparece, mientras que  $\beta_i$  expresa el número de conjuntos en  $I(G_i)$  donde el vértice  $v_i$  aparece, de esta forma  $i(G_i) = \alpha_i + \beta_i$ .

El primer par  $(\alpha_1, \beta_1)$  es  $(1, 1)$  ya que para el subgrafo inducido  $G_1 = \{v_1\}$ ,  $I(G_1) = \{\emptyset, \{v_1\}\}$ . Si sabemos el valor para  $(\alpha_i, \beta_i)$  para cualquier  $i < n$ , y como el siguiente grafo inducido  $G_{i+1}$  se construye de  $G_i$  agregando el vértice  $v_{i+1}$  y la arista  $\{v_i, v_{i+1}\}$ , no es difícil

ver que el par  $(\alpha_{i+1}, \beta_{i+1})$  se construye de  $(\alpha_i, \beta_i)$  aplicando la siguiente ecuación de recurrencia:

$$\alpha_{i+1} = \alpha_i + \beta_i \quad ; \quad \beta_{i+1} = \alpha_i \quad (1)$$

Las series  $(\alpha_i, \beta_i)$ ,  $i=1, \dots, n$ , es construida en base a la recurrencia (1) y conduce a calcular  $i(G_i) = \alpha_i + \beta_i$ , para  $i=1, \dots, n$ . De este modo, el cálculo de  $i(G)$  se basa en el cálculo incremental de  $i(G_i)$ ,  $i=1, \dots, n$ . Si realizamos una búsqueda lineal en el grafo secuencial  $G$  iniciando en un extremo, es decir, empezando en  $v_1$  y moviéndose hacia su vértice adyacente mientras la recurrencia (1) es aplicada, entonces en tiempo lineal sobre el número de vértices, se obtiene la fórmula  $i(P_n) = i(G_n) = \alpha_n + \beta_n = F_{n+2}$ , donde  $F_n$  es el  $n$ -ésimo número de Fibonacci.

Para procesar el número de conjuntos independientes en un camino usaremos lo que llamamos thread de cálculo, o simplemente, thread. Un thread es una secuencia de pares  $(\alpha_i, \beta_i)$ ,  $i=1, \dots, n$  utilizada para calcular el número de conjuntos independientes sobre un camino de  $n$  vértices.

**Caso B:**

Sea  $G = (V, E)$  un árbol. Al recorrer  $G$  en profundidad se construye un árbol enraizado, cuyo nodo raíz es cualquier vértice  $v \in V$  que fue el nodo inicial para empezar la búsqueda en profundidad. Denotamos con  $(\alpha_v, \beta_v)$  el par asociado con el nodo  $v$  ( $v \in G$ ). Calculamos  $i(G)$  mientras recorremos  $G$  en post-orden.

**Algoritmo 1.** Conteo de Conjuntos Independientes en árboles

Count\_Ind\_Sets\_trees( $G$ )

Entrada:  $G$  - un grafo árbol

Salida: El número de Conjuntos Independientes de  $G$

Proceso: Recorrer  $G$  en post-orden, y cuando se visita un nodo  $v \in G$ , asignar:

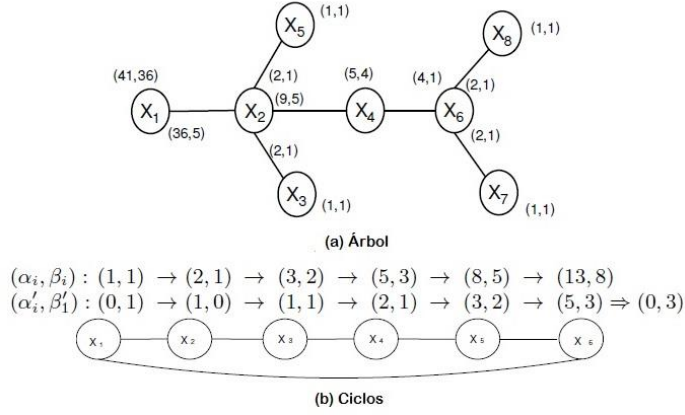
1.  $(\alpha_v, \beta_v) = (1, 1)$  si  $v$  es un nodo hoja en  $G$ .

2. Si  $v$  es un nodo padre con una lista de nodos hijos asociados, es decir,  $u_1, u_2, \dots, u_k$  son los nodos hijos de  $v$ . Ya que hemos visitado todos los nodos hijos, entonces cada par  $(\alpha_{u_j}, \beta_{u_j})$ ,  $j=1, \dots, k$  ha sido determinado basado en la recurrencia (1). Entonces, sea  $\alpha_v = \prod_{j=1}^k \alpha_{u_j}$  y  $\beta_v = \prod_{j=1}^k \beta_{u_j}$ . Note que este paso incluye el caso cuando  $v$  tiene solo un nodo hijo.

3. Si  $v$  es el nodo raíz de  $G$  entonces regresar  $(\alpha_v + \beta_v)$ .

Fin





**Fig. 1.** Contar Conjuntos Independientes en árboles y ciclos

**Ejemplo 1.** Si  $G = \{(x_1, x_2), (x_2, x_3), (x_2, x_4), (x_2, x_5), (x_4, x_6), (x_6, x_7), (x_6, x_8)\}$  es un árbol, consideramos la búsqueda en post-orden y sea  $x_1$  el nodo raíz del árbol. El número de conjuntos independientes en cada nivel del árbol se muestra en la Fig. 1(a). El algoritmo `Count Indep_Sets_trees` regresa  $\alpha_{x_1} = 41$ ,  $\beta_{x_1} = 36$  y el número total de conjuntos independientes es:  $i(G) = 41 + 36 = 77$ .

El procedimiento anterior para contar el número de conjuntos independientes sobre una estructura árbol  $G$ , trabaja en tiempo  $O(n+m)$  el cual, es el tiempo necesario para recorrer  $G$  en post-orden.

**Caso C:**

Otro caso de una topología básica es cuando  $G = (V, E)$ ,  $n=m=|V| = |E|$  es un ciclo simple, es decir, cada vértice en  $V$  tiene grado dos. En este caso, el ciclo puede ser descompuesto como:  $G = G' \cup \{c_m\}$ , donde  $G' = (V, E')$ ,  $E' = \{c_1, \dots, c_{m-1}\}$ ,  $G'$  es un camino de  $n$  vértices, y  $c_m = \{v_m, v_1\}$  es una arista de retroceso del ciclo simple  $G$ .

Observe que cada conjunto independiente de  $G$  es un conjunto independiente de  $G'$ , es decir,  $I(G) \subseteq I(G')$  ya que  $G$  tiene una arista más que  $G'$ . De esta forma, si  $S \in I(G')$  y  $v_1 \in S$  y  $v_m \in S$  entonces  $S$  no es un conjunto independiente de  $G$ . Entonces,  $I(G)$  puede ser construido a partir de  $I(G')$  eliminando aquellos conjuntos independientes que contienen los vértices:  $v_1$  y  $v_m$ , lo que es expresado como:

$$i(G) = i(G') - |\{S \in I(G') : v_1 \in S \wedge v_m \in S\}| \quad (2)$$

Para contar conjuntos independientes en un ciclo simple, podemos usar dos threads o hilos, uno de ellos para calcular  $i(G')$  y el otro thread para calcular  $|\{S \in I(G') : v_1 \in S \wedge v_m \in S\}|$ .

$v_m \in S\}$ . Este último valor se puede calcular fijando sobre  $I(G')$  los conjuntos independientes donde  $v_1$  es implicado, el cual se realiza calculando un thread  $(\alpha'_i, \beta'_i)$ ,  $i=1, \dots, m$  donde el par  $(\alpha'_1, \beta'_1) = (0, 1)$ , considerando en esta forma solo los conjuntos independientes de  $I(G')$  donde  $v_1$  aparece. Aplicamos (1) para calcular las nuevas series:  $(\alpha'_i, \beta'_i)$ ,  $i=2, \dots, m$ . Para considerar solo los conjuntos independientes donde  $v_m$  aparece, del par final  $(\alpha'_m, \beta'_m)$ , se considera sólo el par  $(0, \beta'_m)$ .

En el conteo siguiente, se denota con  $\rightarrow$  la aplicación de la recurrencia (1) sobre  $(\alpha_i, \beta_i)$  para obtener  $(\alpha_{i+1}, \beta_{i+1})$ . Y si expresamos las nuevas series en términos de números de Fibonacci, tenemos que  $(\alpha'_1, \beta'_1) = (0, 1) = (F_0, F_1) \rightarrow (\alpha'_2, \beta'_2) = (1, 0) = (F_1, F_0) \rightarrow (\alpha'_3, \beta'_3) = (1, 1) = (F_2, F_1), \dots, (\alpha'_m, \beta'_m) = (F_{m-1}, F_{m-2})$ , y el valor para el par final  $(\alpha'_m, \beta'_m) = (0, \beta'_m)$  es  $(0, F_{m-2})$ , entonces  $|\{S \in I(G') : v_1 \in S \wedge v_m \in S\}| = 0 + \beta'_m = F_{m-2}$ . Entonces,  $i(G) = i(G') - |\{S \in I(G') : v_1 \in S \wedge v_m \in S\}| = \alpha'_m + \beta'_m - \beta'_m = F_{m+2} - F_{m-2}$ . De este modo, se infiere el siguiente teorema.

**Teorema 1.** Si  $G$  es un ciclo simple con  $n$  vértices entonces el número de conjuntos independientes de  $G$ , expresado en términos de números de Fibonacci, es:  $i(G) = F_{n+2} - F_{n-2}$ .

**Ejemplo 2.** Sea  $E = \{c_i\}_{i=1}^6 = \{\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\}, \{x_4, x_5\}, \{x_5, x_6\}, \{x_6, x_1\}\}$  el conjunto de aristas de un ciclo simple  $G = (V, E)$ . Sea  $G' = (V, E')$  donde  $E = E' \cup \{c_6\}$ .  $G'$  es  $G$  sin la arista  $c_6$ .  $G'$  es un camino de 6 vértices, entonces  $i(G') = F_{6+2} = 21$ . Mientras el valor para  $|\{S \in I(G') : v_1 \in S \wedge x_6 \in S\}|$  es  $F_{6-2} = F_4 = 3$ . Entonces,  $i(G) = 21 - 3 = 18$ , el cálculo se muestra en la Fig. 1 (b).

Todas estas topologías de grafos (casos A, B y C) representan patrones básicos de grafos que se pueden reconocer y procesar para calcular su número de conjuntos independientes en tiempo lineal. Los procesos presentados anteriormente pueden aplicarse al procesar cualquier grafo acíclico o ciclos simples que encontramos como parte de grafos más complejos. En efecto, en De Ita y Tovar (2006) se ha presentado un algoritmo de tiempo polinomial para calcular  $i(G)$  cuando  $G$  tiene composiciones lineales de los patrones arriba mencionados. Podemos ahora preguntar si existe una familia de grafos cíclicos conectados cuyo número de conjuntos independientes puede ser calculado eficientemente, en la siguiente sección mostramos algunas familias de grafos que cumplen este requerimiento.

## 4 Reconocimiento de ciclos embebidos

Sea  $G = (V, E)$  un grafo conectado con  $n = |V|$ ,  $m = |E|$  y tal que  $\Delta(G) \geq 2$ . Para reconocer más patrones de grafos que permitan el cálculo eficiente de  $i(G)$ , presentamos una propuesta para calcular  $i(G)$  sobre grafos outerplanar simples. Para este caso,

introducimos conceptos acerca de la descomposición de un grafo mediante su conjunto de ciclos embebidos.

Si una búsqueda en profundidad (abreviada como *dfs*) se aplica sobre  $G$ , iniciando la búsqueda, por ejemplo, con el vértice  $v_r \in V$  de grado mínimo, y seleccionando entre diferentes vértices potenciales el visitar primero el vértice con grado mínimo y con valor mínimo en su etiqueta como segundo criterio, obtenemos un único grafo  $G'$  resultado de la búsqueda en profundidad (al conjunto resultante de todos los posibles grafos de la búsqueda en profundidad), los cuales denotaremos como  $G' = dfs(G)$ . Este procedimiento *dfs* también construye un único árbol  $T_G$  con  $v_r$  como el nodo raíz.

En tiempo  $O(m+n)$ , *dfs* permite detectar si  $G$  tiene ciclos o no, y las aristas que forman cada ciclo. Las aristas en  $T_G$  se llaman aristas de árbol, mientras que las aristas en  $E(G) \setminus E(T_G)$  se les llama aristas de retroceso. Sea  $e \in E(G) \setminus E(T_G)$  una arista de retroceso, la unión del camino en  $T_G$  entre los puntos extremos de  $e$  con la misma arista  $e$  forma un ciclo simple, tal ciclo se llama un ciclo básico (o fundamental) de  $G$  con respecto a  $T_G$ . Cada arista de retroceso  $e = \{x, y\}$  delimita el camino máximo contenido en el ciclo básico del que es parte. Llamaremos a tal camino máximo, el camino interno del ciclo fundamental. Se asume que  $x$  se visita primero que  $y$  durante la *dfs*, así,  $x$  es el vértice inicial y  $y$  es el vértice final de la arista de retroceso.

Sea  $C = \{C_1, C_2, \dots, C_i\}$  el conjunto de ciclos fundamentales encontrados durante la búsqueda en profundidad. Note que la combinación del procedimiento para árboles y el procesamiento de ciclos (ec. 2) puede ser aplicada para calcular  $i(G)$  si  $G$  es un grafo donde la búsqueda en profundidad genera un árbol y un conjunto de ciclos fundamentales independientes.

Si dos ciclos básicos diferentes  $C_i$  y  $C_j$  de  $C$  tienen aristas comunes entonces decimos que ambos ciclos se intersectan, es decir,  $C_i \Delta C_j$  forman un nuevo ciclo, donde  $\Delta$  denota la operación de diferencia simétrica entre el conjunto de aristas en ambos ciclos. En efecto, si  $C_i \Delta C_j = (E(C_i) \cup E(C_j)) - (E(C_i) \cap E(C_j))$  forma un ciclo compuesto entonces  $C_i$  y  $C_j$  se intersectan. Si dos ciclos no se intersectan decimos que son independientes, es decir, dos ciclos independientes  $C_i, C_j$  cumplen que  $(E(C_i) \cap E(C_j)) = \emptyset$ .

En un grafo outerplanar  $G_o$ , los ciclos de  $G_o$  se pueden considerar como ciclos embebidos unos dentro de otros, ver Fig. 2a. Para reconocer cuándo dos ciclos  $C_i$  y  $C_j$  se pueden expresar como ciclos embebidos, usamos la operación de or-exclusivo (*xor*). Dado dos ciclos intersectados  $C_i$  y  $C_j$ , decimos que  $C_i$  está embebido en  $C_j$  si:

- 1)  $V(C_i) \subset V(C_j)$ : los vértices de  $C_i$  son un subconjunto de los vértices de  $C_j$ .
- 2)  $|E(C_i) - E(C_j)| = 1$  : Existe solo una arista desde  $C_i$  la cual no es arista de  $C_j$ .
- 3)  $C_i \text{ xor } C_j = C_k$  : siendo  $C_k$  un nuevo ciclo distinto a  $C_i$  y  $C_j$  y donde *xor* es la operación or-exclusivo ente las aristas de los ciclos.

Si los ciclos  $C_i$  y  $C_j$  cumplen las tres condiciones previas, decimos que  $C_i$  está embebido en  $C_j$ . Mientras que  $C_j$  es un ciclo que encierra a  $C_i$ .

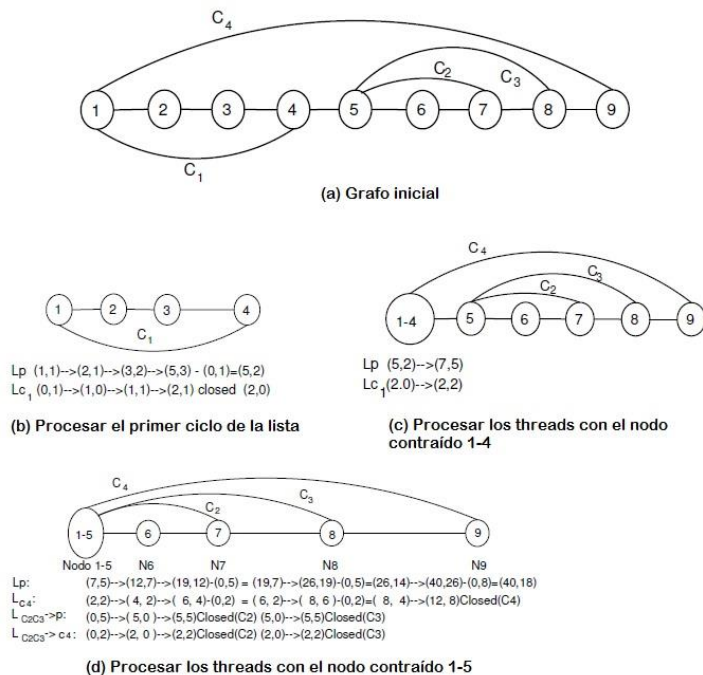


Fig. 2. Cálculo de  $i(D)$ , siendo  $D$  una lista maximal de ciclos embebidos

## 5 Procesamiento de ciclos embebidos

Embeber un objeto en otro es una característica transitiva entre los ciclos embebidos. Si  $C_i$  es embebido en  $C_j$ , y  $C_j$  es embebido en  $C_k$ , entonces  $C_i$  es embebido en  $C_k$ . Por otro lado, Si  $C_i$  y  $C_j$  son dos ciclos independientes, es decir,  $(E(C_i) \cap (E(C_j))) = \emptyset$ , pero existe un ciclo  $C_k$  tal que  $C_i$  y  $C_j$  están embebidos en  $C_k$ , por ejemplo,  $(V(C_i) \subset (V(C_k)))$  y  $(V(C_j) \subset (V(C_k)))$ , entonces decimos que  $(C_k, C_j, C_i)$  es una tupla de ciclos embebidos.

Una lista maximal de ciclos embebidos  $D = (C_1, C_2, \dots, C_k)$  es una tupla de ciclos tal que para  $i < k$ ,  $C_{i+1}$  está embebido en  $C_i$ ,  $i=1, \dots, k-1$ , o existe  $C_j$  en la tupla con  $j < i \leq k$  tal que  $C_i$  está embebido en  $C_j$ . En una lista maximal de ciclos embebidos  $D = (C_1, C_2, \dots, C_k)$  los ciclos son ordenados colocando primero el ciclo más externo seguido por su ciclo interno, así hasta llegar a  $C_k$ , el cual es el ciclo más interno del conjunto de ciclos embebidos. Note que una lista maximal de ciclos embebidos es también un grafo que denotamos por  $D$ .

El cálculo de  $i(C_i)$  sigue el caso (C) para un ciclo simple, descrito en una sección previa. La recurrencia (1) se aplica en los pares actuales  $(\alpha_i, \beta_i)_{L_c} \rightarrow (\alpha_i + \beta_i, \alpha_i)_{L_c}$  cuando un nuevo vértice  $v_{i+1}$  de  $D$  es visitado.

Cada vez que un vértice inicial de  $C_i$  es visitado, el par asociado  $(0, \beta_i)$  se asocia al hilo de ejecución  $L_{C_i}$ , donde  $\beta_i$  es el valor del segundo componente del par  $(\alpha_i, \beta_i)$  asociado a  $L_p$ , ver Fig. 2(b) y 2(d).

Cuando el cálculo llega al vértice final  $v$  de un ciclo  $C_j$  con pares correspondientes  $(\alpha_v, \beta_v)_{C_j}$ , entonces el par  $(0, \beta_v)$  es sustraído a todos los threads de cómputo o de cálculo. Después, el thread de cálculo  $L_{C_j}$  se cierra y deja de estar presente en el cálculo de  $i(D)$ . Cuando un ciclo  $C_j$  ha sido calculado,  $C_j$  puede ser contraído en un solo vértice  $v_{C_j}$ , y el par  $(\alpha_{C_j}, \beta_{C_j})$  se asocia a  $v_{C_j}$ , que es el par resultante al procesar el ciclo  $C_j$ .

El proceso continua con el procesamiento de todos los ciclos en  $D$  hasta llegar al vértice final  $v_f$  de  $D$ . Si  $(\alpha_f, \beta_f)$  es el par asociado a  $v_f$ , entonces  $i(D) = \alpha_f + \beta_f$ .

El siguiente ejemplo muestra como calcular  $i(D)$ , cuando  $D$  es una lista maximal de ciclos embebidos.

**Ejemplo 3.** En la Fig. 2(a), mostramos la lista  $D$  de entrada de ciclos embebidos. En la Fig. 2(b), ya que el primer ciclo de la lista es calculado, entonces dos threads de cálculo se forman. El camino de  $D$  se visita en una forma lineal, y al mismo tiempo la recurrencia (1) se aplica en el par actual de threads, ver Fig. 2(c). Finalmente, la Fig. 2(d) muestra el proceso final de cálculo en todos los threads activos dando como resultado que  $i(D) = 40 + 18 = 58$ .

## 6 Conclusiones

Calcular el número de conjuntos independientes de un grafo  $G$ , denotado como  $i(G)$ , es un problema #P-completo para grafos de grado 3 o superior. En este artículo se presentó un algoritmo eficiente para calcular  $i(D)$ , siendo  $D$  una lista maximal de ciclos embebidos.

Los resultados de nuestra propuesta pueden adaptarse para contar conjuntos independientes en otras estructuras topológicas de grafos, que contengan a los ciclos embebidos como una de sus partes.

## Referencias

- Dahllöf, V. y Jonsson, P. (2002), *An algorithm for counting maximum weighted independent sets and its applications*, *Proc. of SODA'02 thirteenth annual ACM-SIAM Symp. on Discrete Algorithms*, pp. 292-298.
- De Ita, G. y Tovar, M. (2006), *Applying Counting Models of Boolean Formulas to Propositional Inference*, *Advances in Computer Science and Engineering*, Vol.19.

- Greenhill, C. (2000), *The complexity of counting colourings and independent sets in sparse graphs and hypergraphs*. *Computational Complexity*, vol.9 No. 1, pp. 52-72.
- Johnson, D. (1982), *The NP-Completeness Column. An Ongoing Guide*. *Jour. Of Algorithms*, vol. 3, pp. 89-99.
- Okamoto, Y., Uno, T. y Uehara, R. (2005), *Counting the number of independent Sets in chordal graphs*, *Proc. of the 31st Int. WS on Graph-Theoretic Concepts in Computer Science (WG 2005)*, *Lecture Notes in Computer Science*, vol. 3787, pp. 433-444.
- Roth, D. (1996), *On the hardness of approximate reasoning*. *Artificial Intelligence* vol. 82, pp. 273-302.
- Russ, B. (2001), *Randomized Algorithms: Approximation, Generation, and Counting, Distinguished dissertations*. Springer
- Vadhan, S. (2001), *The Complexity of Counting in Sparse, Regular, and Planar Graphs*, *SIAM Journal on Computing*, vol. 31, No. 2, pp. 398-427.

# Capítulo 6

## Modelo de Datos de Consumo Eléctrico para una Función de Optimización

Pérez Camacho Blanca Nydia, González Calleros Juan Manuel

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

blancanydia.perezcc@mail.com, juanmanuel.gonzalez@correo.buap.mx

**Resumen.** En el presente trabajo se presenta la propuesta de solución para el diseño de un sistema de demanda de energía, siguiendo la metodología de desarrollo que se propone para el diseño del sistema es necesario obtener o generar datos para probar el sistema de demanda. La modelación de los datos se hace a partir de una simulación y de datos reales obtenidos en una investigación del consumo eléctrico de una casa disponibles en la red es a partir de estas dos fuentes de datos es que se generan los datos sintéticos a utilizar. La selección de los dispositivos, para los cuales se generan los datos sintéticos, son identificados a partir de la disponibilidad de su función de distribución de probabilidades de uso y de los dispositivos que han sido monitoreados en un estudio para caracterizar el consumo eléctrico de una casa.

**Palabras Clave:** Modelado de datos, datos sintéticos, simulación de datos.

### 1 Introducción

Un sistema de demanda se diseña he implementa con el objetivo de analizar y controlar que el consumo eléctrico de un escenario se realice de manera eficiente. Los objetivos de un sistema de demanda son: reducir el consumo eléctrico, costos, picos de voltaje o el de maximizar el confort de los usuarios (Rahim et al., 2016). El diseño de un sistema de demanda se puede hacer desde distintas líneas de investigación, como interfaces humano-computadora, arquitecturas de monitoreo y control, técnicas de inteligencia artificial para realizar la búsqueda del consumo ad hoc al escenario, seguridad en telecomunicaciones, protocolos de comunicación, redes, almacenamiento, y web de las cosas. Y para poder llevar acabo un sistema de este tipo es necesario contar con datos que provean al sistema con lo fundamental para poder probar su funcionamiento. Una vía común de abastecimiento de los datos es mediante la implementación de una arquitectura hardware

que genere dichos datos, otra manera de generar los datos es mediante la creación de estos de estos. Una técnica que sirve para generar los datos sintéticos de consumo es implementando un simulador. El simulador puede basarse en la función de distribución de probabilidades, generando así datos que parten de datos reales. Es posible complementar los datos resultantes de un simulador con datos reales de consumo del monitoreo de un conjunto de dispositivos de una casa. Por lo cual el propósito principal de este documento ahonda en la explicación de la creación de un modelo de datos de consumo, los cuales son necesario para poder probar el funcionamiento de un sistema de demanda cuando no se cuenta con una infraestructura física que provea de dichos datos.

En la sección 2 se presenta la revisión de la literatura que hay alrededor del presente trabajo. La metodología de desarrollo del modelado de los datos que forman el espacio de búsqueda se muestra en la sección 3, en la sección 4 se aborda el tema de como se realizó la implementación. En las secciones 5 y 6, se tratan de los resultados y de las conclusiones respectivamente.

## 2 Preliminares

El desarrollo de los sistemas de demanda se puede hacer desde distintos enfoques, como lo son el uso de centros de control (Kakran & Chanana, 2018), consumo (Rasheed et al., 2016), costos (Fauvel, Claveau, Chevrel, & Fiani, 2018), arquitectura de sensores (Javaid, Naseem, et al., 2017) y técnicas de control (Rajasekaran, Manikandaraj, & Kamaleshwar, 2017). La metodología para el desarrollo de un sistema de demanda puede tomarse desde dos enfoques: el flujo de datos (Chen et al., 2017) y la arquitectura ciberfísica (Matei, Anton, Scholze, & Cenedese, 2017). A partir de ambos enfoques se puede proponer una metodología que encaja en la arquitectura de un sistema de gestión de energía para el hogar (Huang, Tian, & Wang, 2015). Como parte del desarrollo de un sistema de demanda es necesario identificar el tipo de datos que se manejan, y que saber que los datos de consumo son valores continuos. Los valores de consumo se encuentran en un rango determinado por el propio dispositivo que hace el consumo (Javaid, Hussain, et al., 2017). Así como también, es necesario identificar las variables que determinan el consumo eléctrico (Jahn et al., 2010), los tipos de dispositivos, el modelo matemático del consumo (Nadeem, Javaid, Malik, & Iqbal, 2018), y la técnica de búsqueda a usar (Javaid, Naseem, et al., 2017). Para poder implementar un sistema de manejo de energía es necesario contar con datos de consumo, pero cuando no se cuenta con una arquitectura hardware que los proporcione entonces se recurre a la creación de datos sintéticos mediante la definición de un modelo de datos (Ojeda Zújar, Díaz Cuevas, Prieto Campos, & Álvarez Francoso, 2013). En las siguientes secciones se relata tanto la metodología, así como también el como se hizo la implementación para la obtención de un modelo de datos para un sistema de demanda.



### 3 Metodología

La metodología que se sigue para el desarrollo de la propuesta de solución de un sistema de demanda se divide en tres fases (véase la Fig. 1). La primera fase sirve para generar los datos sintéticos, la segunda sirve para la obtención de los perfiles de consumo y la implementación de una técnica de búsqueda, y la tercera fase genera un reporte de recomendación de estado de los dispositivos.

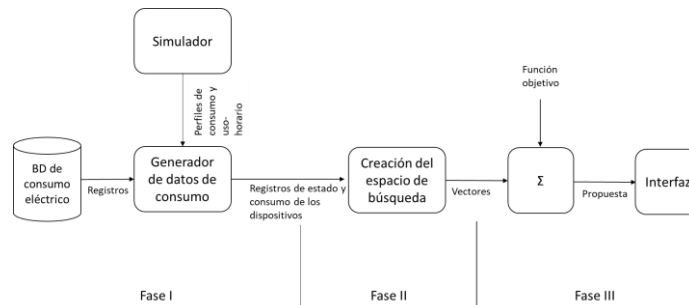


Figura 1. Esquema general de la metodología

La primera fase está conformada por dos bloques de generación de datos. El primer bloque es un simulador que genera datos de consumo y perfiles de uso de cuatro dispositivos. El segundo bloque genera datos de consumo de seis dispositivos seleccionados a partir de una base de datos que contiene el reporte de consumo de un conjunto de dispositivos de una casa. Los datos que se generan sirven para construir vectores de consumo por hora de diez dispositivos, los cuales en conjunto forman el espacio de búsqueda.

La segunda fase, comprende desde que se tienen los datos para los vectores que conforman el espacio de búsqueda hasta la implementación de una técnica metaheurística. Al espacio de búsqueda se aplica una técnica metaheurística, para este caso en específico un algoritmo genético, para realizar la búsqueda del estado óptimo de los dispositivos de acuerdo con la función objetivo. La función objetivo para este caso de aplicación se define a partir de un modelo matemático que busca minimizar el consumo eléctrico. Por consiguiente, lo que se está buscando es el estado óptimo del conjunto de dispositivos que minimizan el consumo eléctrico de entre un espacio de posibles estados de los dispositivos.

En la tercera fase se considera la salida de la implementación de la técnica metaheurística para ser presentada visualmente a través de una interfaz.

### 3.1 Paso 1. Generación de Datos de Consumo

La metodología de desarrollo de la primera fase (véase la Fig. 2) parte de la obtención de las funciones de distribución de probabilidades de uso de cuatro dispositivos y de una tabla de dispositivos que contiene: datos de consumo mínimo y máximo, tiempos de uso frecuentes y el número frecuente que se tienen en una casa de estos dispositivos. Para cada uno de los dispositivos se obtienen las coordenadas de la FDP de aquellos puntos donde cambia el comportamiento de la gráfica. La gráfica representa el estado del dispositivo en un periodo de 24 horas. Una vez obtenidas las coordenadas, estas sirven para alimentar un simulador de uso. El método de desarrollo del simulador parte de los parámetros con los cuales se cuentan, para este caso en específico se ha implementado el método Monte Carlo usando la técnica de aceptación-rechazo. Como salida de la simulación se obtienen dos matrices, una de horarios de uso y la otra del consumo eléctrico del dispositivo en el horario especificado por la primera matriz. También, como parte de la primera fase se desarrolló un segundo módulo que se enfoca en proveer de datos de consumo a partir de una base de datos que se encuentra disponible en la red. La base de datos contiene datos reales de consumo de un conjunto de dispositivos que son comúnmente ubicados en una casa habitación, los datos de esta base de datos fueron recabados durante un periodo de 30 meses.

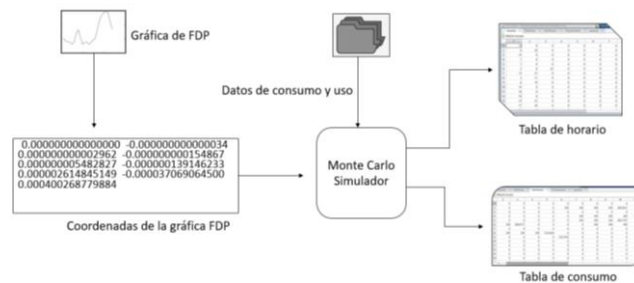


Figura 2. Diagrama de la metodología de la Paso 1

### 3.2 Paso 2. Generación de Vectores

Para el diseño del segundo módulo de la fase uno (véase la Fig. 3) se parte de la limpieza y preprocesamiento de los datos de la base de datos de consumo. Una vez identificados los campos que conforman a la base de datos, son eliminados aquellos que no sirven a la finalidad para la cual se desea implementar el bloque. Después, la tabla es procesada para obtener el consumo eléctrico total por hora, una vez finalizado el procesamiento se selecciona de manera aleatoria uno de los registros obtenidos. Del registro se obtiene de los campos correspondientes la hora y el consumo total. Posteriormente, se crea una tabla donde cada uno de los campos corresponde a los

dispositivos que se han seleccionado (15 dispositivos, de los cuales cuatro son los simulados y cada uno de estos aparece tantas veces como los especificados por el numero promedio que se llega a encontrar en una casa, y el resto de los dispositivos son los especificados en la tabla de consumos). La tabla se llena con las posibles combinaciones de estado de los 15 dispositivos. Para cada uno de los campos que corresponde a los dispositivos simulados, se seleccionan de manera aleatoria uno de los perfiles generados en el simulador, usando los perfiles de horario y de consumo se identifican tanto el consumo como el horario al cual se encuentra relacionado, de acuerdo a estos dos parámetros es que en la tabla de combinaciones en el campo al que corresponde al dispositivo se coloca en uno si es que el dispositivo genero algún consumo en la hora identificada, en caso contrario es colocado en cero; y este procedimiento se hace para cada uno de los cuatro dispositivos. Esta tabla sirve de base para la creación de una nueva tabla cuyos campos son los mismo que los de esta primera, y el llenado de esta se hace colocando los consumos que haya realizado el dispositivo siempre y cuando el estado del dispositivo de la primera tabla sea 1, en caso contrario se considera como consumo cero. Después, esta segunda tabla que se ha generado es filtrada dejando solo aquellos registros donde la suma de los consumos esté en un índice de  $\pm 5\%$  con respecto al consumo total seleccionado de la base procesada de consumos. La segunda tabla ya filtrada es la salida de la fase uno, esta sirve para representar los vectores de consumo necesarios para la fase dos.

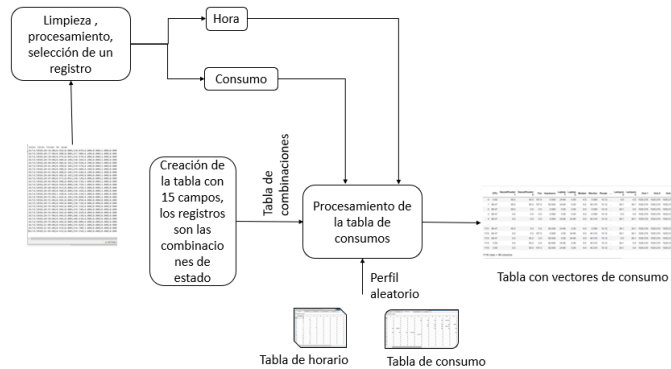


Figura 3. Diagrama de la metodología del paso 2

### 3.3 Paso 3. Sistema de Búsqueda

La tabla resultante de la Fase II se usa para conformar los vectores de consumo de los dispositivos, conformando así vectores de 15 parámetros. El conjunto de vectores de consumo son los que forman al espacio de búsqueda. En el espacio de búsqueda que se ha formado es en el cual se implementa la técnica de búsqueda heurística, para este caso específico se implementará un algoritmo genético, y la función objetivo se conceptualiza

para minimizar el consumo eléctrico (véase la Fig. 4). La conceptualización de la función objetivo parte de la identificación de los tipos de dispositivos (interrumpibles, no-interrumpibles, flexibles y no -flexibles) que se van a contemplar, así como también del comportamiento en el uso de estos dispositivos, ambos datos son los que permiten una formalización matemática. La formalización matemática comprende una serie de ecuaciones que representan al conjunto de cada tipo de dispositivos y a como calcular el consumo eléctrico de cada tipo, para finalmente encontrar la función que expresa el objetivo que se quiere cumplir con el sistema de manejo de energía, que en este caso es el de minimizar el consumo eléctrico. Como salida de la implementación de la técnica heurística se genera un vector de estado de los dispositivos que cumple con la función objetivo de minimización de consumo. El vector de estado de los dispositivos se presenta visualmente ante el usuario como una propuesta del estado en cual debería encontrarse los dispositivos para que él pueda tomar la decisión de seguir la recomendación o dejarlos en el estado en el que se están.

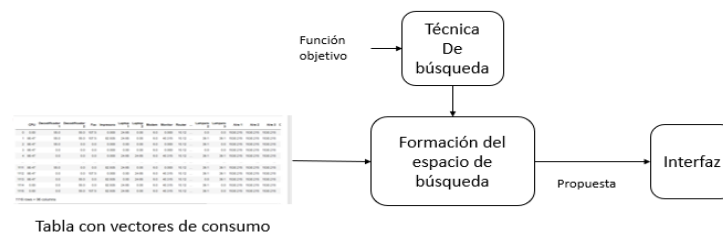


Figura 4. Diagrama de la metodología del paso 3

## 4 Implementación

Para el desarrollo de la propuesta de solución, en la fase uno parte del análisis de la función de distribución de probabilidades de cuatros dispositivos, y de la obtención de una base de datos que se genero a partir del almacenamiento del consumo eléctrico de un conjunto de dispositivos, los dispositivos fueron monitoreados por 30 meses.

El bloque del simulador de la primera fase se desarrolló en MatLab implementando la técnica de aceptación-rechazo (véase la Fig. 5) del Método Monte Carlo. Para el desarrollo de este simulador fue necesario contar con la función de distribución de probabilidades (FDP) del dispositivo. A partir de las gráficas de la FDP se obtuvieron los datos de las coordenadas de los puntos característicos (los máximos y mínimos) a lo largo de las 24 horas del día. Estos datos junto con los rangos de consumo eléctrico, los tiempos de uso de cada uno de estos dispositivos (de acuerdo con el área geográfica para el cual se desea llevar acabo la simulación) y el número frecuente de estos dispositivos que llega a haber en una casa son ingresados como parámetros.

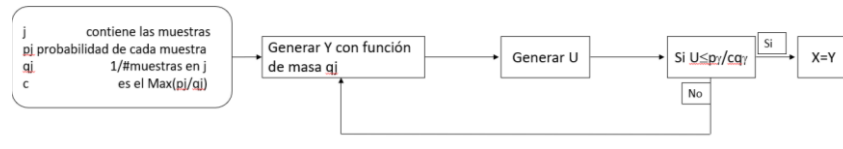


Figura 5. Técnica de Aceptación-Rechazo del Método Monte Carlo

El simulador se usó para cuatros dispositivos diferentes de uso doméstico: secadora, extractor de aire, aire acondicionado y ventilador. A partir de la simulación se genera para cada uno de los cuatros dispositivos dos matrices (véase la Fig. 6). Cada una de las filas de cada una de las matrices representa un experimento. Lo que se almacena en cada una de las filas de la primera matriz son los horarios en los cuales es probable se pueda llegar a encender el dispositivo; y, en la segunda matriz se almacena el consumo que se haya generado para cada uno de los horarios en los que se encendió.

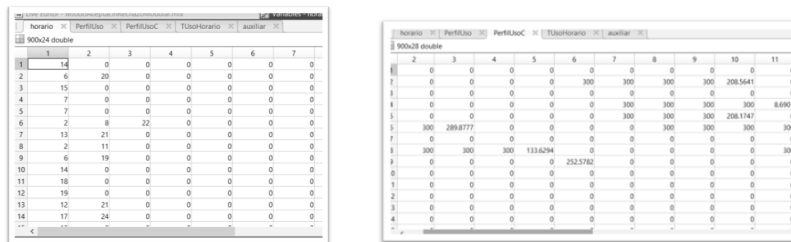


Figura 6. Matrices de horario y consumo

El segundo módulo de la fase uno, es el módulo que provee de los datos de consumo de seis dispositivos especificados a partir de la base de datos de consumo, así como del consumo total real que se puede tener en una hora determinada (Fig. 7). Este segundo modulo se desarrolló en el lenguaje Python, se partió del análisis de la base de datos la cual se limpió y proceso para calcular los consumos que se hacen por hora (Fig. 8).

Se crearon nuevas tablas con los campos correspondientes a cada uno de los 15 dispositivos, primero se calcularon las posibles combinaciones de estado de los dispositivos (Fig. 9), después se obtuvieron los consumos y estado de los cuatro dispositivos en la simulación para la hora para la cual se desea generar el espacio de búsqueda. Una vez colocados los consumos en aquellos campos en los que el estado de uso sea de encendido se calculan el consumo total y se filtran solo aquellos que cumplen con el criterio de estar en el rango de  $\pm 5\%$  del consumo total real reportado en la base de datos de consumo (Fig. 10).

date_	time_	global_Active	global_Reactive	voltage	global_intensity	sub_metering_1	sub_metering_2	sub_metering_3	
0	16/12/2006	17:24:00	4.216	0.418	234.84	18.4	0	1	17
1	16/12/2006	17:25:00	5.360	0.436	233.63	23.0	0	1	16
2	16/12/2006	17:26:00	5.374	0.496	233.29	23.0	0	2	17
3	16/12/2006	17:27:00	5.368	0.502	233.74	23.0	0	1	17
4	16/12/2006	17:28:00	3.666	0.528	235.68	15.8	0	1	17
...	...	...	...	...	...	...	...	...	...
1044501	13/12/2008	21:34:00	0.426	0.076	242.27	1.8	0	0	0
1044502	13/12/2008	21:35:00	0.424	0.076	242.10	1.8	0	0	0
1044503	13/12/2008	21:36:00	0.422	0.076	241.73	1.8	0	0	0
1044504	13/12/2008	21:37:00	0.422	0.078	242.56	1.8	0	0	0
1044505	13/12/2008	21:38:00	0.422	0.078	242.61	1.8	0	0	0

1044506 rows x 9 columns  
**Figura 7.** Tabla de los datos de consumo

date_	time_	global_Active	sub_metering_1	sub_metering_2	sub_metering_3	hour_	minute_	suma_global	suma_sub1	suma_sub2	suma_sub3	
0	16/12/2006	17:59:00	2.472	0	0	17	17	59	152.024	0	19	66
1	16/12/2006	18:59:00	4.224	0	1	18	18	59	217.632	0	403	10
2	16/12/2006	19:59:00	3.214	0	0	17	19	59	204.014	0	86	10
3	16/12/2006	20:59:00	3.308	0	0	17	20	59	196.114	0	0	10
4	16/12/2006	21:59:00	3.454	0	2	17	21	59	183.388	0	25	10
...	...	...	...	...	...	...	...	...	...	...	...	...
17409	13/12/2008	16:59:00	2.750	2	1	17	16	59	240.660	1483	30	71
17410	13/12/2008	17:59:00	1.510	0	0	0	17	59	165.316	657	2	61
17411	13/12/2008	18:59:00	1.376	0	0	0	18	59	79.740	0	0	0
17412	13/12/2008	19:59:00	3.216	0	1	17	19	59	135.606	0	33	31
17413	13/12/2008	20:59:00	0.394	0	0	0	20	59	88.362	0	0	6

17414 rows x 12 columns  
**Figura 8.** Tabla con consumo por hora

CPU	Decodificador_1	Decodificador_2	Fax	Impresora	Laptop_1	Laptop_2	Modem	Monitor	Router	Lampara_2	Lampara_3	Aire_1	Aire_2	Aire_3	Calefactor_1	Cal
0	0	1	1	1	0	1	0	1	0	1	0	0	0	1	1	0
1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
2	1	1	0	0	0	0	1	0	1	1	0	1	1	1	1	1
3	1	0	0	0	0	0	1	0	1	0	0	1	1	1	1	1
4	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1111	1	1	0	0	1	1	0	1	0	1	1	1	1	1	1	1
1112	1	0	0	1	0	0	1	1	1	1	1	1	1	1	1	0
1113	1	0	1	0	1	0	1	1	1	1	0	1	1	1	1	0
1114	0	0	1	0	1	1	0	1	1	1	1	0	1	1	1	0
1115	0	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1

1116 rows x 16 columns  
**Figura 9.** Tabla de las combinaciones de estado de los dispositivos

CPU	Decodificador_1	Decodificador_2	Fax	Impresora	Laptop_1	Laptop_2	Modem	Monitor	Router	Lampara_3	Aire_1	Aire_2	Aire_3	Calefactor_1	Cal
0	0.00	55	55	107.5	0.000	24.65	0.00	6	0.000	10.12	0.0	1530.215	1530.215	1530.215	
1	88.47	55	55	107.5	82.935	24.65	0.00	6	40.315	10.12	39.1	1530.215	1530.215	1530.215	
2	88.47	55	0	0.0	0.000	0.00	0.00	6	0.000	10.12	0.0	1530.215	1530.215	1530.215	
3	88.47	0	0	0.0	0.000	0.00	0.00	6	0.000	10.12	0.0	1530.215	1530.215	1530.215	
4	88.47	0	0	0.0	0.000	24.65	24.65	6	40.315	10.12	39.1	1530.215	1530.215	1530.215	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1111	88.47	55	0	0.0	82.935	24.65	0.00	6	0.000	10.12	39.1	1530.215	1530.215	1530.215	
1112	88.47	0	0	107.5	0.000	0.00	24.65	6	40.315	10.12	39.1	1530.215	1530.215	1530.215	
1113	88.47	0	55	0.0	82.935	0.00	24.65	6	40.315	10.12	39.1	1530.215	1530.215	1530.215	
1114	0.00	0	55	0.0	82.935	24.65	0.00	6	40.315	10.12	0.0	1530.215	1530.215	1530.215	
1115	0.00	0	68	107.5	82.935	24.65	0.00	6	40.315	10.12	0.0	1530.215	1530.215	1530.215	

**Figura 10.** Tabla con consumos

## 5 Resultados

Para las simulaciones de cada uno de los cuatro dispositivos se define el número de experimentos por simulación, donde se toma la desviación de un 5%. El número de experimentos necesarios para cada uno de los dispositivos se muestran en la Tabla 1.

Tabla 1  
Número de experimentos de cada uno de los dispositivos

Dispositivo	Experimentos
Secador de ropa	100
Extractor de aire	186 – 189
Aire acondicionado	418
Calefactor	387

Las matrices de consumo y de horario permiten generar graficas de uso de los dispositivos, de las gráficas se obtienen los porcentajes de encendido (secadora de ropa: 6-7%, extractor de aire: 13-14%, aire acondicionado: 13-14%, y calefactor: 15-16%).

A partir de la base de datos de consumo de una casa son procesados sus 1044506 registros usando lo descrito en la metodología como en la implementación de la Fase II, dando como resultado 32768 registros de consumo (Fig. 11). Con los registros de consumo se conforman los vectores de consumo del espacio de búsqueda.

	CPU	Decodificador 1	Decodificador 2	Fax	Impresora	Laptop 1	Laptop 2	Modem	Monitor	Router	...	Lampara 3	Aire 1	Aire 2	Aire 3	Calefactor 1	Calefactor 2	
0	0.00	55	55	107.5	0.000	24.65	0.00	6	0.000	10.12	...	0.0	1530.215	1530.215	1530.215	0		
1	88.47	55	55	107.5	82.935	24.65	0.00	6	40.315	10.12	...	39.1	1530.215	1530.215	1530.215	1		
2	88.47	55	0	0.0	0.000	0.00	0.00	6	0.000	10.12	...	0.0	1530.215	1530.215	1530.215	1		
3	88.47	0	0	0.0	0.000	0.00	0.00	6	0.000	10.12	...	0.0	1530.215	1530.215	1530.215	1		
4	88.47	0	0	0.0	0.000	24.65	24.65	6	40.315	10.12	...	39.1	1530.215	1530.215	1530.215	0		
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1111	88.47	55	0	0.0	82.935	24.65	0.00	6	0.000	10.12	...	39.1	1530.215	1530.215	1530.215	1		
1112	88.47	0	0	107.5	0.000	0.00	24.65	6	40.315	10.12	...	39.1	1530.215	1530.215	1530.215	0		
1113	88.47	0	55	0.0	82.935	0.00	24.65	6	40.315	10.12	...	39.1	1530.215	1530.215	1530.215	0		
1114	0.00	0	55	0.0	82.935	24.65	0.00	6	40.315	10.12	...	0.0	1530.215	1530.215	1530.215	0		

Figura 11. Tabla que contiene los registros filtrados de acuerdo con el consumo total

## 6 Conclusiones

A partir de las simulaciones es posible observar en sus resultados que tanto el extractor como el calefactor tienen comportamientos similares, lo que quiere decir que la frecuencia de uso y la manera en la cual son utilizados es muy similar. Cuando no se cuenta con una infraestructura que proporcione los datos de consumo necesarios para implementar y probar técnicas de búsqueda del estado de los dispositivos idóneos para lograr reducir el consumo eléctrico es posible crear un conjunto de datos sintéticos a partir

de un simulador y de una base de datos de consumo generales. Logrando genera un espacio de búsqueda con datos que, aunque son sintéticos fueron calculados partir de datos reales.

## Referencias

- Chen, S., Liu, T., Gao, F., Ji, J., Xu, Z., Qian, B., ... Guan, X. (2017). Butler, Not Servant: A Human-Centric Smart Home Energy Management System. *IEEE Communications Magazine*, 55(2), 27–33. <https://doi.org/10.1109/MCOM.2017.1600699CM>
- Fauvel, C., Claveau, F., Chevrel, P., & Fiani, P. (2018). A flexible design methodology to solve energy management problems. *International Journal of Electrical Power and Energy Systems*, 97(November 2017), 220–232. <https://doi.org/10.1016/j.ijepes.2017.11.005>
- Huang, Y., Tian, H., & Wang, L. (2015). Demand response for home energy management system. *International Journal of Electrical Power and Energy Systems*, 73, 448–455. <https://doi.org/10.1016/j.ijepes.2015.05.032>
- Jahn, M., Jentsch, M., Prause, C. R., Pramudianto, F., Al-Akkad, A., & Reiners, R. (2010). The energy aware smart home. *2010 5th International Conference on Future Information Technology (FutureTech), IEEE.*, 1–8. <https://doi.org/10.1109/FUTURETECH.2010.5482712>
- Javaid, N., Hussain, S., Ullah, I., Noor, M., Abdul, W., Almogren, A., & Alamri, A. (2017). Demand Side Management in Nearly Zero Energy Buildings Using Heuristic Optimizations. *Energies*, 10(8), 1131. <https://doi.org/10.3390/en10081131>
- Javaid, N., Naseem, M., Rasheed, M. B., Mahmood, D., Khan, S. A., Alrajeh, N., & Iqbal, Z. (2017). A new heuristically optimized Home Energy Management controller for smart grid. *Sustainable Cities and Society*, 34(July), 211–227. <https://doi.org/10.1016/j.scs.2017.06.009>
- Kakran, S., & Chanana, S. (2018). Energy Scheduling of Smart Appliances at Home under the Effect of Dynamic Pricing Schemes and Small Renewable Energy Source. *International Journal of Emerging Electric Power Systems*, 1–12. <https://doi.org/10.1515/ijeeps-2017-0187>
- Matei, O., Anton, C., Scholze, S., & Cenedese, C. (2017). Multi-layered data mining architecture in the context of Internet of Things. *Proceedings - 2017 IEEE 15th International Conference on Industrial Informatics, INDIN 2017*, 1193–1198. <https://doi.org/10.1109/INDIN.2017.8104943>
- Nadeem, Z., Javaid, N., Malik, A. W., & Iqbal, S. (2018). Scheduling appliances with GA, TLBO, FA, OSR and their hybrids using chance constrained optimization for smart homes. *Energies*, 11(4), 1–30. <https://doi.org/10.3390/en11040888>
- Ojeda Zújar, J., Díaz Cuevas, M. del P., Prieto Campos, A., & Álvarez Francoso, J. I. (2013). Línea de costa y sistemas de información geográfica: modelo de datos para



- la caracterización y cálculo de indicadores en la costa andaluza. *Investigaciones Geográficas*, (60), 37–52. <https://doi.org/10.14198/ingeo2013.60.02>
- Rahim, S., Javaid, N., Ahmad, A., Khan, S. A., Khan, Z. A., Alrajeh, N., & Qasim, U. (2016). Exploiting heuristic algorithms to efficiently utilize energy management controllers with renewable energy sources. *Energy and Buildings*, 129, 452–470. <https://doi.org/10.1016/j.enbuild.2016.08.008>
- Rajasekaran, R. G., Manikandaraj, S., & Kamaleshwar, R. (2017). Implementation of Machine Learning Algorithm for Predicting User Behavior and Smart Energy Management, 24–30.
- Rasheed, M., Javaid, N., Awais, M., Khan, Z., Qasim, U., Alrajeh, N., ... Javaid, Q. (2016). Real Time Information Based Energy Management Using Customer Preferences and Dynamic Pricing in Smart Homes. *Energies*, 9(7), 542. <https://doi.org/10.3390/en9070542>

# Capítulo 7

## Desarrollo de la competencia lectora con apoyo de las TIC

Adelina Escobar Acevedo, Josefina Guerrero García

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

adelina.escobar@alumno.buap.mx, josefina.guerrero@correo.buap.mx

**Resumen.** En el presente trabajo se hace una revisión de cómo se han utilizados las TIC como apoyo en el desarrollo de la competencia lectora. La lectura es una de las habilidades que por sus componentes cognitivos y metacognitivos conlleva beneficios a lo largo de la vida del ser humano. Es de interés de los gobiernos que sus ciudadanos tengan acceso a desarrollar sus competencias lectoras y éstas a su vez representan un indicador de la educación del país. Esta revisión del uso de la tecnología en la lectura nos indica los logros alcanzados y los retos por vencer en su asistencia.

**Palabras Clave:** Comprensión lectora, TIC, Lectura.

### 1 Introducción

La lectura es una actividad fundamental en el ámbito escolar; por ello la Secretaría de Educación Pública (SEP) considera objetivos durante cada nivel educativo, desde preescolar hasta la educación media superior. El primer logro esperado está relacionado con el lenguaje y las Tecnologías de la Información y Comunicación (TIC) (ver Tabla 1): Se expresa y comunica correctamente, oralmente y por escrito, con confianza y eficacia... sabe identificar ideas clave en textos para inferir conclusiones... tiene capacidad de análisis y síntesis; sabe argumentar, es crítica, reflexiva, curiosa, creativa y exigente... es competente y responsable en el uso de las TIC (SEP, 2016).

Durante la infancia, específicamente, los primeros años de educación elemental, los niños aprenden a leer mediante la decodificación de símbolos y la comprensión de palabras. Acorde a Simmons & Kameenui (1998) uno de cada seis niños enfrentara algún problema durante el proceso de aprender a leer dentro de los primeros tres años de escuela. Si no se resuelven en ese mismo periodo, existe una probabilidad alta de que el estudiante sufra consecuencias durante el resto de su vida.

Tabla 1 Logros esperados en el ámbito de Lenguaje y Comunicación en cada nivel educativo (Secretaría de Educación Pública (SEP), 2016)

<i>Ambitos</i>	<i>Al término de preescolar</i>	<i>Al término de la primaria</i>	<i>Al término de la secundaria</i>	<i>Al término de la Educación Media Superior</i>
Lenguaje y Comunicación	Expresa emociones, gustos e ideas en su lengua materna	Comunica sentimientos, sucesos e ideas en su lengua materna y en español, tanto de su forma oral como escrita; se comunica en inglés en actividades simples y cotidianas; usa las TIC para satisfacer su curiosidad y expresar ideas.	Se comunica en español y en su lengua materna con eficacia oralmente y por escrito, con múltiples propósitos y en contextos diversos; describe en inglés necesidades, acontecimientos y aspiraciones; usa las TIC para realizar investigaciones y para comunicarse	Argumenta con eficacia y se expresa en español y en su lengua materna con claridad, de forma oral y escrita; se comunica en inglés con fluidez y naturalidad; utiliza las TIC para obtener, procesar, interpretar información y comunicar con eficacia.

La era digital en la que vivimos actualmente ha modificado nuestro estilo de vida incluyendo los hábitos de lectura. Las TIC se han integrado paulatinamente a los medios educativos generando un cambio significativo, han permitido la creación de estrategias de enseñanza y aprendizaje ofreciendo una motivación activa; los Ambientes Virtuales de Aprendizaje se han expandido como una alternativa viable que aprovecha la tecnología en lugares remotos mediante el uso de plataformas educativas como Moodle o Blackboard apostando por el aprendizaje autónomo y autorregulado. Sin embargo, la aplicación de Inteligencia Artificial en la educación ha sido insipiente, principalmente mediante realidad virtual o Sistemas Tutores Inteligentes (Almaraz Rodríguez & Navarro Rodríguez, 2017).

## 2 Preliminares

La OCDE (Organización para la Cooperación y el Desarrollo Económicos) afirma que particularmente la competencia lectora es uno de los predictores más confiables del bienestar socioeconómico, incluso más que el total de años de estudio (OCDE, 2012). Es por ello que la evaluación lectora se encuentra incluida en la prueba PISA, examen estandarizado a nivel internacional que se aplica cada tres años a jóvenes escolarizados alrededor de los 15 años de edad.

Las pruebas PISA nos han demostrado desde hace casi veinte años que los estudiantes mexicanos tienen un bajo desarrollo de la competencia lectora, y prácticamente se ha mantenido la misma puntuación todos estos años (418 puntos en promedio), con una tendencia a la baja para el año 2018 (ver Figura 1). Los estudiantes mexicanos se encuentran en el nivel II de los seis que existen en PISA. En este nivel los estudiantes sólo pueden localizar una o más piezas de información; pueden reconocer la idea principal de

un texto; entender o construir significados en partes limitadas del texto, y pueden hacer inferencias de bajo nivel (OCDE, 2018).

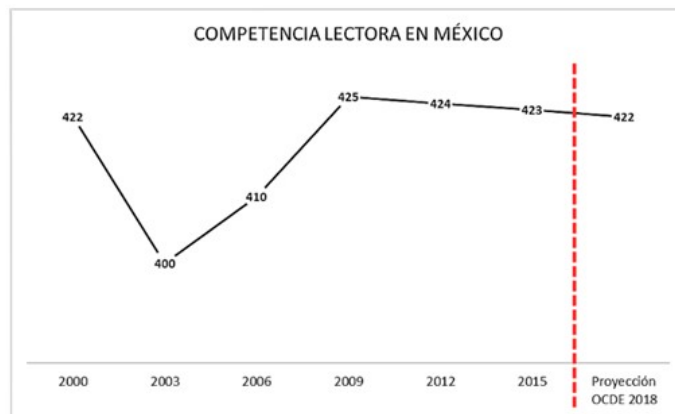


Figura 1 Competencia lectora en México acorde a la OCDE, (OCDE, 2018).

La competencia lectora incluye varias habilidades cognitivas como la decodificación, conocimiento de vocabulario, estructuras gramaticales, características lingüísticas y conocimiento del mundo. A su vez, involucra algunas habilidades metacognoscitivas como el uso de estrategias para comprensión de textos (Hernández Padilla & Bazán Ramírez, 2016).

De acuerdo con PISA “la competencia lectora es la capacidad de un individuo de comprender, emplear, reflexionar e interesarse en textos escritos con el fin de lograr metas propias, desarrollar sus conocimientos y su potencial personal y participar en la sociedad (Díaz & Flores, 2010) .

A través del Instituto de Estadística, la UNESCO indicó en 2017 que más de 617 millones de niños y adolescentes no están alcanzando los niveles mínimos de competencia en lectura y matemáticas. En América Latina y el Caribe 64% de niños y jóvenes que no alcanzan los niveles de lectura se encuentran escolarizados. (UNESCO, 2017)

Los esfuerzos no se concentran únicamente en el desarrollo de la habilidad, en México, el INEGI lleva a cabo un levantamiento llamado MOLEC (Módulo sobre Lectura), cuyo objetivo es generar información estadística sobre el comportamiento lector de la población adulta que reside en áreas de 100 mil y más habitantes y con ello conocer las principales características de la práctica de la lectura en el país. MOLEC considera libros, revistas, periódicos, historietas, páginas de internet, foros y blogs como material de lectura. El levantamiento realizado en febrero de 2019 indicó que el porcentaje de población alfabeta que leyó algún material de los considerados por MOLEC fue de 74.8%, lo que indica un decremento de 10 puntos porcentuales con respecto al levantamiento del 2015 en donde el porcentaje fue de 84.2%. (INEGI, 2019).

### 3 La lectura en la era digital

El Marco Común Europeo de referencia para las lenguas: aprendizaje, enseñanza, evaluación (Consejo de Europa, 2002) indica que, para leer el alumno debe saber (p. 89):

- Percibir el texto escrito (destrezas visuales).
- Reconocer la escritura (destrezas ortográficas).
- Identificar el mensaje (destrezas lingüísticas).
- Comprender el mensaje (destrezas semánticas).
- Interpretar el mensaje (destrezas cognitivas).

La UNESCO realiza un análisis de la lectura desde la incorporación de la tecnología móvil. Indican que los recursos impresos son costosos y de difícil acceso, sobre todo en zonas remotas y poco pobladas. Sin embargo, la lectura digital ha logrado establecerse exitosamente. Una de sus ventajas más palpables es la capacidad acceder a textos digitales mediante la fuerte expansión del uso de teléfonos móviles. Los libros son escasos a nivel mundial, pero los teléfonos móviles están en todos lados, 6 mil millones de personas tienen acceso a un teléfono móvil, las redes de telefonía celular alcanzan más del 95% de la población mundial. Un tercio de los lectores usan los teléfonos para leerles a los niños, la lectura digital es viable (Piedras, 2015).

Se ha indicado que la educación tradicional contrasta con los entornos cada vez más llamativos fuera de las aulas, programas televisivos, videojuegos, publicidad, invierten en grandes campañas cuyo objetivo es mantener la atención del usuario. Los detractores indican que el abuso de la tecnología a edades tempranas tiene efectos perjudiciales en el desarrollo emocional del niño dado que será incapaz de manejar frustración ya que la tecnología provee soluciones inmediatas. Sin embargo, como menciona Odorico (2004) indica que el docente no puede ser un mero observador ante el avance que está teniendo lugar en la sociedad en cuanto a nuevas tecnologías multimediales.

El uso de recursos tecnológicos puede propiciar situaciones idóneas de aprendizaje; los libros digitales, las bibliotecas virtuales, o las revistas electrónicas, proporcionan recursos informáticos inmensos.

El uso de Entornos o Ambientes Virtuales de Aprendizaje son lugares ideales para la generación y aprendizaje de nuevos conocimientos; esta perspectiva enfatiza en la colaboración de todos los actores involucrados contribuyendo a un aprendizaje más efectivo (Duarte D., 2003).

Existen apps educativas de uso comercial que se convierten en el instrumento más atractivo para mejorar la comprensión desde las edades tempranas: Rapids (<https://rapids.amazon.com/>); watterpad ([https://www.watterpad.com/?locale=es\\_ES](https://www.watterpad.com/?locale=es_ES)); Doted (<http://appteca.apps4citizens.org/doted-aumenta-interes-velocidad-lectura-jovenes/>); Hooked (<http://appteca.apps4citizens.org/hooked-leer-millennials-formato-chat/>). Apps para leer libros como Kindle (Amazon), iBooks (Apple), Kobo (<https://www.kobo.com/>), Aldiko ([www.aldiko.com](http://www.aldiko.com)), Audible ([www.audible.com](http://www.audible.com)).

Con esto los ambientes educativos tratan de incluirse más lúdicos, más accesibles usando en ocasiones las mismas estrategias e inclusive promueven sistemas de recompensa.

## 4 Tecnología como apoyo a la lectura

Entre los trabajos realizados para apoyar la lectura por medio de la tecnología destacan los siguientes:

**Reading with Ross**, fue un programa lanzado en 1995 que requería la presencia de un asistente quien apoyaba a pequeños grupos de 4 o 5 niños durante media hora 4 días cada semana con el objetivo de incentivar la lectura y lograr conciencia fonética (B. W. Wise y Olson, 1995). Los resultados fueron incrementos significativos en las habilidades de lectura, sin embargo la dependencia a un asistente humano, el grupo reducido y los requerimientos tecnológicos propiciaron el desarrollo de otras herramientas.

En español, un tutor inteligente desarrollado en Chile obtuvo buenos resultados al aplicarlo a niños detectados con problemas en lectura, contiene ejercicios de vocabulario y juegos de dictado, sin embargo expresan que la motivación inicial se perdió conforme avanzaron las lecciones debido a la poca variedad de material disponible (Strasser et al., 1997).

**Baldi**, un agente virtual precursor que enseña vocabulario a niños con problemas auditivos. Tiene la ventaja de hacer su rostro transparente para permitir ver al alumno los movimientos de dientes y lengua al emitir los sonidos. Como resultado, los niños aprendieron alrededor de 70 palabras diarias y recordaban 39 después de 4 semanas (Barker, 2003).

**Ms Redwrite** es un tutor virtual dentro de libros interactivos que proporciona tips y explicaciones para ayudar a los niños a encontrar respuestas mientras aprenden sonidos, lectura, pronunciación y comprensión. Así aprenden a reconocer palabras, leer fluidamente y entender lo que leen. (B. Wise et al., 2005)

La mayoría de los trabajos en que se usa tecnología como apoyo a la lectura se centran en las fases iniciales del aprendizaje, por ello están enfocados a niños. Sin embargo, resalta la tesis doctoral “Leyendo con robots” 2018 cuyo objetivo es proveer a adultos mayores con un agente conversacional capaz de realizar preguntas sobre metáforas encontradas en novelas literarias. Se plantea como una alternativa para mantener lúcido el cerebro a través de la lectura retardando la aparición de la demencia. (Parde, 2018).

Entre los trabajos citados, se encuentran varias aplicaciones de Procesamiento de Lenguaje Natural necesarios para escalar las investigaciones o mejorar la interacción hombre máquina. Es una tendencia incluir recursos tecnológicos para mantener una interacción fluida a fin de mantener el interés por parte del usuario.

Se ha considerado el uso de Sistemas Tutores Inteligentes (STI) que apoyen en la competencia lectora; los STI para lenguas extranjeras se basan en los desarrollos de la Inteligencia Artificial, principalmente, en lo que compete a la utilización de técnicas de

Comprensión de Lenguaje Natural y de Generación de Lenguaje Natural. Un STI es un programa para la enseñanza-aprendizaje basado en el computador cuya finalidad última es la facilitación de los procesos de aprendizajes personalizados y autónomos. Estos STI se enfocan en apoyo para aprender vocabulario y realizar conversaciones (Ferreira, Salcedo, Kotz, y Barrientos, 2012), aunque la mayoría están enfocados a áreas de matemáticas y programación.

## 5 Conclusiones

El uso de las TIC en la educación ha sido una estrategia favorecedora y motivadora; se busca que la lectura digital sea un apoyo para mejorar la competencia lectora. La literatura nos reporta que se emplean diferentes recursos como los libros digitales, las aplicaciones móviles o en la web para fomentar esta competencia. El apoyo a la lectura mediante la tecnología se ha ido fortaleciendo conforme el desarrollo de la misma lo permite, actualmente existen desarrollos que permiten determinar el interés del usuario mediante aplicaciones de reconocimiento de expresiones faciales y movimientos oculares. A su vez, la interactividad se está logrando mediante la incorporación de agentes conversacionales que permiten el reconocimiento de lenguaje hablado y la transformación de texto a audio. Incluso la incorporación de sistemas adaptativos dan pie a flexibilidad y personalización. Una variante es la comprensión lectora de un segundo idioma que permita alcanzar los niveles de suficiencia de idioma indicados en el Marco Común Europeo.

Las TIC favorecen un aprendizaje significativo, su uso permite desarrollar estrategias de lectura, de creatividad y de comunicación; aún hace falta considerar la incorporación de otras tecnologías como las tecnologías del lenguaje (que tienen como fin último la comprensión del lenguaje humano por una computadora) con el fin de mantener una retroalimentación constante entre usuario y tecnología.

## Referencias

- Almaraz Rodríguez, O. D., y Navarro Rodríguez, M. (2017). Un Sistema Tutor Inteligente que Favorece el Aprendizaje de la Lengua Escrita (U. P. de Durango, Ed.). Durango.
- Barker, L. J. (2003). Computer-Assisted Vocabulary Acquisition: The CSLU Vocabulary Tutor in Oral-Deaf Education. *Journal of Deaf Studies and Deaf Education*, 8(2), 187–198. <https://doi.org/10.1093/deafed/eng002>.
- Consejo de Europa. (2002). Marco Común Europeo de Referencia para las Lenguas. Centro Virtual Cervantes. Recuperado de [http://cvc.cervantes.es/ensenanza/biblioteca\\_ele/marco/default.htm](http://cvc.cervantes.es/ensenanza/biblioteca_ele/marco/default.htm).

- Díaz, M. A., y Flores, G. (2010). México en Pisa 2009. Recuperado de [http://www.inee.edu.mx/images/stories/Publicaciones/Estudios\\_internacionales/PISA\\_2009/Completo/pisa2009.pdf](http://www.inee.edu.mx/images/stories/Publicaciones/Estudios_internacionales/PISA_2009/Completo/pisa2009.pdf).
- Duarte D., J. (2003). Ambientes De Aprendizaje: Una Aproximación Conceptual. *Estudios Pedagógicos (Valdivia)*, (29), 97–113. <https://doi.org/10.4067/s0718-07052003000100007>
- Ferreira, A., Salcedo, P., Kotz, G. y Barrientos, F. (2012). La Arquitectura de ELE-TUTOR: Un Sistema Tutorial Inteligente para el Español como Lengua Extranjera. *Revista signos*, 45(79), 102-131. <https://dx.doi.org/10.4067/S0718-09342012000200001>
- Hernández Padilla, E., y Bazán Ramírez, A. (2016). Efectos contextuales, socioeconómicos y culturales, sobre los resultados de México en Lectura en PISA 2009. *REICE. Revista Iberoamericana Sobre Calidad, Eficacia y Cambio En Educación*, 14.2(2016), 79–95. <https://doi.org/10.15366/reice2016.14.2.005>.
- INEGI. (2019). Comunicado De Prensa Núm. 190 / 19 23 De Abril De 2019. 17. Recuperado de: [https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2019/EstSociodemo/MOLEC2018\\_04.pdf](https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2019/EstSociodemo/MOLEC2018_04.pdf).
- OCDE. (2012). PISA 2009 Results: What Students Know and Can Do. In PISA 2009 Results: What Students Know and Can Do. <https://doi.org/10.1787/9789264188716-ar>.
- OCDE. (2018). Marco teórico de lectura PISA 2018. Recuperado de: [https://www.mecd.gob.es/inee/dam/jcr:2f1081a1-c1e4-4799-8a49-9bc589724ca4/marco\\_teorico\\_lectura\\_2018\\_esp\\_ESP.pdf](https://www.mecd.gob.es/inee/dam/jcr:2f1081a1-c1e4-4799-8a49-9bc589724ca4/marco_teorico_lectura_2018_esp_ESP.pdf)
- Odorico, A. (2004). Marco teórico para una robótica pedagógica. *Revista de Informática Educativa y Medios Audiovisuales* Vol. 1(3), 1(3), 34–46. Recuperado de: <http://laboratorios.fi.uba.ar/lie/Revista/Articulos/010103/A4oct2004.pdf>.
- Parde, N. (2018). Reading with robots: A platform to promote cognitive exercise through identification and discussion of creative metaphor in books. University of North Texas.
- Piedras, E. (2015). Reflexiones sobre lectura digital. *La Lectura En La Era Móvil: Un Estudio Sobre La Lectura Móvil En Los Países En Desarrollo. Una Mirada Desde México*, 4–5. Recuperado de: <http://www.unesco.org/new/fileadmin/MULTIMEDIA/FIELD/Mexico/miradamexico.pdf>.
- Secretaría de Educación Pública (SEP). (2016). Los Fines de la Educación en el Siglo XXI. *Perfiles Educativos*, 38(154), 216–225. Recuperado de: <https://www.academica.org/sebastian.urquijo/57.pdf>
- Simmons, D. C., y Kameenui, E. J. (1998). What Reading Research Tell Us About Children With Diverse Learning Needs. University of Oregon.
- Strasser, K., Rosas, R., Bazueta, R., Zamorano, R., Moulían, M., Nussbaum, M., y Bermeosolo, J. (1997). Desarrollo de un Sistema Tutorial Inteligente para el Apoyo de la Enseñanza de la Lectura Inicial. *Psykhe (Santiago)*, 6(1), 65–79.



- UNESCO. (2017). Instituto de estadística de unesco. Instituto de Estadística de Unesco, (46), 1–5. Recuperado de: <http://uis.unesco.org/sites/default/files/documents/fs46-more-than-half-children-not-learning-2017-sp.pdf>.
- Wise, B., Cole, R., Vuuren, S. Van, Schwartz, S., Snyder, L., Ngampatipatpong, N., y Tuantranont, J. (2005). Learning to Read with a Virtual Tutor: Foundations to Literacy. *Interactive Literacy Education: Facilitating Literacy Learning Environments through Technology*.
- Wise, B. W., y Olson, R. K. (1995). Computer-based phonological awareness and reading instruction. *Annals of Dyslexia*, 45(1), 97–122. <https://doi.org/10.1007/BF02648214>.

# Capítulo 8

## Actualización de la Interfaz de Control y Monitoreo del Piano Cósmico

Tonatiuh García Chávez<sup>1</sup>, Arturo Fernández Téllez<sup>1,2</sup>, Guillermo Tejeda Muñoz<sup>2</sup>

<sup>1</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

<sup>2</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias Físico Matemáticas

tonaspiuck@gmail.com, Arturo.Fernandez.Tellez@cern.ch,  
Guillermo.Tejeda.Munoz@cern.ch

**Resumen.** En el presente trabajo se muestra la programación de una interfaz gráfica (GUI) y actualización de una parte del sistema de adquisición de datos del piano cósmico (primera patente gestionada desde la BUAP, IMPI 311700). El aporte que se tiene dentro del trabajo de tesis es un primer acercamiento a los elementos con los que se trabajará en el proyecto, como lo son detectores de partículas, dispositivos electrónicos programables y protocolos de comunicación, entre otros.

**Palabras Clave:** detectores de partículas, GUI, multihilo.

### 1 Introducción

El trabajo que se pretende llevar a cabo en el proceso del doctorado tiene como fin la adquisición, manipulación y análisis de datos provenientes de detectores de partículas. Por lo tanto se comenzó haciendo una actualización a un proyecto ya existente, lo cual sirve para adentrarse al tipo de elementos con los que se trabajarán en el proyecto final. Específicamente se contemplan las dos primeras acciones del proyecto de tesis, adquisición y manipulación de datos, en una situación similar, aunque mucho menos demandante computacionalmente hablando a aquella que se presentará en el proyecto de tesis.

La tarea es hacer una actualización a la interfaz del piano cósmico, es decir crear un programa con interfaz gráfica para configurar y mostrar la salida de la recolección de datos provenientes de detectores de radiación. Primeramente, en los preliminares, se expondrá el proyecto del piano cósmico, se dará una breve explicación de lo que son los rayos cósmicos y detectores de partículas, posteriormente en el desarrollo del trabajo se

presentan los requerimientos y características que se deben cumplir en el proyecto además de los resultados, para finalmente exponer las conclusiones del trabajo.

## **2 Preliminares**

### **Piano Cósmico**

El piano cósmico fue un proyecto que surgió para hacer visible lo invisible, para mostrar al público en general, sin importar edades o niveles educativos, cómo el físico de partículas identifica a las llamadas partículas elementales, está diseñado para detectar y hacer que el paso de rayos cósmicos, a través del aparato, se traduzcan en distintas frecuencias auditivas y destellos luminosos. Su aplicación notoria es hacer dinámica la detección de partículas cósmicas por medio de nemotecnias rítmico-melódicas al escuchante. Tal invento fue construido con componentes utilizados por ACORDE, el detector de rayos cósmicos del experimento ALICE (Boletín Buap, 2014).

El piano cósmico es un arreglo de cinco módulos detectores de partículas, el funcionamiento básico es cuando una partícula atraviese algún detector entonces se mostrara una señal luminosa acompañada de una frecuencia musical, además de llevar el conteo de la cantidad de partículas que han pasado por cada detector y mostrarlo al público. Dentro de la electrónica contiene un bloque de discriminación y acondicionamiento de la señal detectada proveniente de cada detector y un bloque de almacenamiento, los cuales no fueron modificados en este trabajo.

En el sitio [http://alicematters.web.cern.ch/?q=ALICE\\_cosmicpiano](http://alicematters.web.cern.ch/?q=ALICE_cosmicpiano) se puede ver en acción el Piano Cósmico, en dueto con el pianista Al Palmer, en el Festival de Jazz de Montreux, Suiza. Se puede apreciar el primer prototipo con cuatro detectores y a un lado una pequeña botonera la cual es la interfaz original para controlar el modo de funcionamiento del piano cósmico, misma que se pretende actualizar en este trabajo.

### **Radiación y Rayos Cósmicos**

Por lo general cuando hablamos de radiación se piensa en un efecto negativo o destructivo. Es de gran importancia enterarse que vivimos en un mundo naturalmente radioactivo, la mayor parte de la radiación recibida por la población proviene de fuentes naturales y es inevitable exponerse a la mayoría de ellas. Por lo tanto, el hombre siempre se ha desarrollado en un ambiente naturalmente radioactivo.

A la radiación proveniente del espacio exterior se le conoce como rayos cósmicos, son partículas cargadas que viajan por el espacio a velocidades cercanas a las de la luz y transportan una cantidad muy grande de energía. No se tiene información muy clara

acerca de cuáles son las fuentes de rayos cósmicos que llegan a la tierra, se conjetura que son “multi-mensajeros” de restos de acontecimientos más energéticos del universo por ejemplo explosión de estrellas (supernovas) o choques de galaxias, sin embargo, los rayos cósmicos ultra energéticos probablemente provengan de algún lugar fuera de nuestra galaxia (Stone et al, 2018, Daniela Blanco Lira, 2018).

Cuando un rayo cósmico llega a la atmósfera se encuentra con moléculas de esta, el choque entre ambas partículas produce una serie de partículas secundarias que conforme avanzan en su recorrido vuelven chocar con más partículas provocando una reacción en cadena, esto se denomina chubasco de partículas o lluvias atmosféricas. En la Figura 1 muestra una simulación de tres diferentes chubascos de partículas, donde cada uno es originado por un rayo cósmico distinto (Hernán Asorey, 2012).

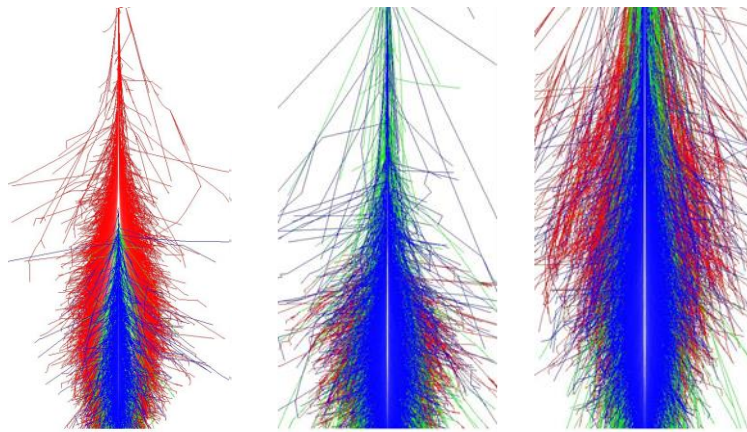


Figura 1. Desarrollo de tres lluvias atmosféricas: en la izquierda un fotón, en medio un protón y a la derecha un núcleo de hierro. Todos verticales y con  $E_p = 5 \times 10^{14}$  eV. Los diferentes colores identifican las tres componentes principales: electromagnética (rojo), muónica (verde) y hadrónica (azul) (Hernán Asorey, 2012).

Cabe recalcar que en este trabajo no se hará análisis o clasificación alguna del tipo o propiedades de la partícula que se ha detectado, será suficiente con reconocer que ha sucedido un evento de donde se ha captado el paso de una partícula.

## Detectores de Partículas

Los detectores de radiación son transductores de energía, que transforman la energía de la radiación en otro tipo de energía que podemos cuantificar fácilmente, por ejemplo: en pulsos eléctricos, emisores de luz, o cambios en las propiedades del material.

Existen distintos tipos de detectores de radiación, en este trabajo se utilizaron plásticos centelladores que son del mismo tipo a aquellos utilizados en el detector ACORDE del experimento ALICE.

### 3 Desarrollo y Resultados

Como se mencionó anteriormente el propósito es actualizar la interfaz del piano cósmico para que sea más amigable con cualquier usuario en cuanto al manejo y monitoreo de los datos recolectados. El piano cósmico tiene dos modos de funcionamiento: single y coincidencia. En el modo single cada detector funciona por separado, es decir independiente de los otros detectores, mientras que, en el modo coincidencia se crea un grupo personalizado de detectores que se encuentran “unidos” y sólo aumentará la cuenta del grupo si es que todos los elementos dentro del grupo han detectado una partícula dentro de una ventana de tiempo, de lo contrario el grupo no aumentará la cuenta. Se tiene que satisfacer las siguientes características:

- Al momento de detectar una partícula emitir luz y sonido
- La luz y sonido emitido debe estar sincronizado, es decir, ejecutarse al mismo tiempo.
- Cada detector tiene su propio sonido y su propia luz.
- Llevar un conteo de eventos independiente para cada detector.
- Para ser válido un incremento de cuenta en modo coincidencia el evento capturado debe suceder dentro en una ventana de tiempo de máximo 100 ns.
- La interfaz debe ser lo más amigable posible para el uso del público.

Primeramente, para la parte electrónica se utilizó un FPGA, esto debido a la velocidad necesaria en el modo coincidencia ya que solo se cuenta con 100 ns máximo para validar o no el evento. La idea consiste en ser lo suficientemente rápido con el fin de ser capaz de detectar estos eventos pero al mismo tiempo mostrar adecuadamente (sincronizando la cuenta, la luz y el sonido) los datos al público. El protocolo que se implementó para la comunicación entre dispositivos fue UART con una velocidad de 115200 bps.

Por otro lado, el dispositivo que se utilizó para programar la interfaz fue una tarjeta RaspberryPi 4. Dicha tarjeta tiene la ventaja de contar con una salida propia para la visualización de la interfaz en cualquier pantalla, cuenta con GPIOs que sirven para una correcta sincronización de los eventos además de conexiones dedicadas al protocolo UART que se programó en la FPGA para la recepción y transferencia de datos.

Finalmente, la interfaz gráfica se programó en Python 3.7. En este punto se tiene un ligero inconveniente, Python no es capaz de manejar el puerto serie y el programa que esté en ejecución de manera concurrente. Lo anterior es un obstáculo ya que al ser una interfaz gráfica donde el flujo de información y acciones no es secuencial si no que es controlada por eventos, los cuales no se sabe cuándo ocurrirán, provoca que la interfaz

gráfica no pueda reaccionar a la interacción con el usuario y como consecuencia de esto la aplicación se congelaba. Lo anterior obviamente hace una interfaz gráfica completamente inservible. La solución: hacer la aplicación multihilo donde un hilo se enfocaba en la lectura del puerto serie, así como la actualización de la interfaz, mientras que otro hilo se encargaba de la interacción con el usuario (Burkhard A. Meier ,2017). En la Figura 2 se muestra un esquema de las partes que involucran al proyecto y su comunicación.

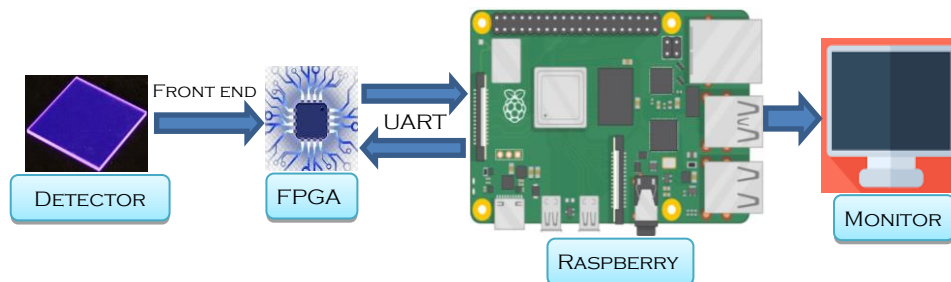


Figura 2. Diagrama a bloques de la conexión entre los dispositivos del sistema.

Teniendo en cuenta los requerimientos del sistema se pasó a la implementación de las partes que conforman el proyecto. Cada detector de partículas tiene su propia cuenta a mostrar dentro de la interfaz. También se establecieron dos grupos independientes que pueden trabajar en modo coincidencia, por ejemplo, en el grupo uno podría tener incluido al detector tres y cuatro mientras que en el grupo dos puede involucrar al detector dos y cinco.

En la Figura 3 se muestra la aplicación final creada. Todos los bloques se muestran al usuario en la interfaz para que pueda ver la cuenta de las partículas detectadas y configurar en modo single o coincidencia. Cada bloque cuenta con una etiqueta que define a que detector corresponde, el valor de la cantidad de partículas que ha registrado y un botón de reset en caso de que se requiera reiniciar la cuenta. En los extremos se muestran los bloques para la configuración en modo coincidencia, cada bloque contiene el título del grupo, una serie de radio-buttons para elegir los detectores que se tomarán en cuenta para el modo coincidencia de ese grupo, muestra la cantidad de eventos validos en modo coincidencia y un botón de reset si se desea reiniciar la cuenta.

En cuanto a la luz y sonido del piano cósmico cada grupo y cada detector tienen su propio color de luz además de su propio sonido. El sonido que emite fue seleccionado con base en las notas de las escalas pentatónicas, tales escalas tienen la peculiaridad de ser agradables al escuchante sin importar el orden en que se toquen lo cual es una propiedad que se explota ya que es aleatorio el orden en que las partículas atraviesan los detectores.



Figura 3.- Captura de la interfaz gráfica creada para el control y monitoreo del piano cósmico

## 4 Conclusiones

El trabajo presentado sirvió como introducción a los aparatos y dispositivos con los que se trabajarán en el tema de tesis, además del mapeo de señales, programación y condiciones que se deben satisfacer en un ambiente real. Ciertamente la implementación electrónica, la cantidad de datos, así como la velocidad y procesamiento de estos, serán parámetros mucho más exigentes y minuciosos en el trabajo final.

Se reporta la actualización de la interfaz de control y monitoreo del piano cósmico. Se cumplió con una correcta sincronización de las señales que el público percibe además de poder ser configurado en sus diferentes modos de operación. Para la comunicación entre los dispositivos se implementó el protocolo de comunicación UART y finalmente, para la interfaz gráfica creada fue necesario un enfoque multihilo con el fin de mostrar un comportamiento adecuado al usuario. Dentro de los eventos de difusión y puertas abiertas de la universidad se puede visitar el proyecto mencionado en este trabajo.

## Referencias

- Boletín BUAP (2014). *Mexicanos dejan huella en el experimento ALICE del CERN*. Recuperado de [http://emas.siu.buap.mx/portal\\_pprd/wb/comunic/piano\\_cosmico\\_primera\\_patente\\_gestionada\\_desde\\_la](http://emas.siu.buap.mx/portal_pprd/wb/comunic/piano_cosmico_primera_patente_gestionada_desde_la)
- Tanabashi, M., Hagiwara, K., Hikasa, K., Nakamura, K., Sumino, Y., Takahashi, F., ... & Antonelli, M. (2018). "Review of particle physics". *Physical Review D*, 98(3), 030001.

- Daniela Blanco Lira (2018). *MUTE: sistema de detección de muones atmosféricos para los experimentos ALICE y MATHUSLA en el LHC del CERN* (Tesis doctoral). Benemérita Universidad Autónoma de Puebla, México.
- Hernán Asorey (2012). *Los Detectores Cherenkov del Observatorio Pierre Auger y su Aplicación al Estudio de Fondos de Radiación* (Tesis Doctoral). Universidad Nacional de Cuyo, Argentina.
- Burkhard A. Meier (2017). *Python GUI Programming Cookbook*. Birmingham, UK: Packt Publishing Ltd.



# Capítulo 9

## Extracción de Características de la Postura Corporal como una Variable Correlacionada con el Nivel de Atención

Alfredo García<sup>1</sup>, Juan Manuel González<sup>1</sup> y Amparo Palomino<sup>2</sup>

<sup>1</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

<sup>2</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Electrónica

alfredo\_amigo18@hotmail.com, jumagoca78@gmail.com y  
ampalomino@gmail.com

**Resumen.** Hoy en día, se han asociado varios factores que intervienen en el comportamiento del nivel de atención de las personas. Estos factores pueden ser internos o externos al contexto en el que se desarrolla el usuario. Este artículo aborda uno de los factores inherentes al comportamiento humano: la postura corporal. A partir de esta variable, se obtiene la extracción de 8 características estadísticas (media, varianza, oblicuidad, curtuosis, desviación estándar, máximo, mínimo y rango), que se procesan matemáticamente para eliminar mediciones fuera de rango y mediciones producidas por errores o ruido presente en el experimento. Para el proceso de visualización, las características extraídas se correlacionan con la media del porcentaje de atención que se obtiene de las ondas cerebrales para finalmente mostrar su forma en gráficos 2D. Este análisis se realiza utilizando herramientas estadísticas que demuestran gráficamente algunas diferencias en el nivel de atención entre un niño con TDAH y un niño sin TDAH. Con los resultados obtenidos en este trabajo y como trabajo futuro, se buscarán patrones de comportamiento y modelos matemáticos que describan el nivel de atención basado en la postura corporal del usuario.

**Palabras Clave:** Nivel de atención, señales cerebrales, postura corporal, diagnóstico de TDAH, Coeficiente de correlación y análisis estadístico.

### 1 Introducción

El TDAH se caracteriza por falta de atención, impulsividad e hiperactividad. Recientemente se ha estimado que afecta al 3,5% de los niños en edad escolar en todo el mundo y se dice que es uno de los trastornos psiquiátricos más comunes entre los jóvenes. Los niños con estos problemas a menudo son impopulares y carecen de amistades

recíprocas, pero no siempre son conscientes de su propia impopularidad. Aunque estos síntomas tienden a disminuir con la edad, al menos el 50% de los niños con TDAH todavía tienen síntomas que disminuyen en la edad adulta. A pesar de la vasta literatura que respalda la eficacia de la medicación estimulante en el tratamiento del trastorno por déficit de atención / hiperactividad (TDAH), varias limitaciones de los tratamientos farmacológicos resaltan la clara necesidad de tratamientos psico-sociales alternativos efectivos. También hay evidencia de intervenciones que involucran tanto a la escuela como a la capacitación de los padres que han resultado en clasificarlas como "tratamientos validados empíricamente" mostrados en (Pascual, Begoña y Buldian, 2010). El trastorno por déficit de atención con hiperactividad (TDAH) es una afección neurobiológica común que afecta a niños en edad escolar. Uno de los síntomas principales es la falta de atención, que es un factor clave del bajo rendimiento académico, especialmente en tareas que requieren mucho tiempo de concentración como se indica en (Asiry, Shen y Calder, 2015). Los niños con trastorno por déficit de atención e hiperactividad (TDAH) experimentan un déficit en los procesos cognitivos responsables de los comportamientos dirigidos a objetivos específicos, conocidos como funcionamiento ejecutivo (FE) descrito en (Weisberg et al, 2014). El mayor desafío para los adultos con trastorno por déficit de atención con hiperactividad (TDAH) es el manejo de la información y las tareas como se menciona en (Sonne y Jensen, 2016).

## **2 Sistemas para medir el nivel de atención**

El estudio de señales fisiológicas como ondas cerebrales, frecuencia cardíaca, temperatura corporal, entre otros, ha revelado grandes avances en los últimos tiempos obteniendo resultados significativos en aplicaciones de diferentes campos de estudio como la medicina, la robótica, la psicología, entre otros. Actualmente existen dispositivos comerciales de bajo costo para obtener la lectura de las señales cerebrales con las cuales es posible conocer el nivel de atención del usuario de una manera poco confiable. El rendimiento de estos dispositivos se limita al software y hardware establecido por el fabricante para una tarea específica, sumado a la falta de precisión en la lectura de las señales, ya que los dispositivos no tienen un sistema robusto para la adquisición y el procesamiento de datos. El rendimiento de estos dispositivos es limitado, ya que su fabricación está orientada a tareas simples o juegos didácticos. Otra causa del bajo rendimiento de estos dispositivos es que son invasivos o intrusivos; Tiaras, cascos, muestras de sangre se utilizan para obtener las señales de las variables fisiológicas del usuario, cualquier error en la calibración podría generar un error en el diagnóstico final.

Para conocer el grado de afectación que el TDAH produce en las personas, es necesario contar con herramientas que puedan proporcionar una retroalimentación del porcentaje de atención al ejecutar una tarea específica. Actualmente existe una variedad de dispositivos comerciales que proporcionan cuantitativamente el nivel de concentración, meditación, relajación y atención al usuario, pero en algunos casos se logran de manera invasiva,

afectando la respuesta del usuario y, en consecuencia, el diagnóstico final. Estos dispositivos generalmente usan una variable fisiológica para inferir los niveles de atención en las personas, son del tipo de usuario único y tienen un costo accesible. El rendimiento de estos dispositivos es limitado ya que tienen restricciones por parte del fabricante con respecto al software y hardware implementado. La adquisición de datos y las velocidades de procesamiento de dispositivos MINDWAVE, EMOTIV EPOC, MUSE, entre otros; tienen retrasos y no pueden obtener una lectura de las variables adquiridas en un tiempo aproximado al tiempo real. Otra desventaja presentada por este tipo de dispositivo es su baja usabilidad y versatilidad en la práctica, ya que el usuario requiere mucho tiempo para que los dispositivos reconozcan las señales fisiológicas que se desean adquirir. Algunos dispositivos tienen una interfaz gráfica diseñada por el fabricante, cuyos comentarios se basan únicamente en la indicación gráfica del nivel de atención. Diversas aplicaciones en áreas como: psicología, educación, negocios, salud, entre otras, requieren un sistema que identifique con precisión el nivel de atención en las personas, y que a su vez proporcione una respuesta instantánea de lo que sucede, así como un diagnóstico final confiable para la toma de decisiones. También es deseable obtener una retroalimentación que aliente al usuario a elevar el nivel de atención al mismo tiempo que ejecuta una tarea específica.

Los sistemas de entrenamiento de bio-retroalimentación fomentan un estado mental o físico específico en un usuario a través de un ciclo cerrado de retroalimentación biológica. Estos sistemas recopilan el estado fisiológico de una persona a través de la detección de hardware, integran este estado en un sistema interactivo basado en computadora y presentan los comentarios para que el usuario pueda trabajar para ajustar su estado (Sonne y Jensen, 2016).

En este trabajo de investigación, proponemos implementar un sistema para medir el nivel de atención en niños con TDAH, generando un análisis de las muestras obtenidas, con el fin de caracterizar el comportamiento de sus ondas cerebrales y obtener patrones estadísticos que nos permitan identificar la presencia de TDAH en estudiantes a través del uso de este dispositivo. El sistema no es invasivo y tiene una interfaz centrada en el usuario, con el objetivo de obtener un diagnóstico final que describa de manera confiable el nivel de atención.

## **2.1 Variables fisiológicas relacionadas con el nivel de atención**

La variedad de técnicas aplicadas en el campo de la investigación para cuantificar el nivel de atención de las personas conduce a un análisis descriptivo que se presenta en este trabajo (Marín, 2014).

A partir de la revisión en la literatura de las variables utilizadas para relacionar el nivel de atención en las personas, se ha encontrado que las ondas cerebrales son las variables fisiológicas con mayor relevancia debido a la relación cognitiva que existe entre el pensamiento y la actividad cerebral. Por lo tanto, su implicación es directa, como se describe en (Saneiro, 2015).

El gráfico de la Fig. 1 muestra la relevancia que tiene cada una de las variables fisiológicas en el nivel de atención de las personas. Este análisis se obtiene del estado del arte de las obras relacionadas.

Dependiendo de la variable fisiológica utilizada, el dispositivo se elige para realizar la adquisición de datos. Dentro de la mayoría de los dispositivos utilizados en la literatura se encuentran: el electroencefalograma, la cámara WEB, los sensores de movimiento, los giroscopios, los electrodos, el mouse, el electrocardiograma, los sensores electroquímicos, el teclado, los transductores, las cámaras y los sensores ópticos, mencionados en (García, 2015).

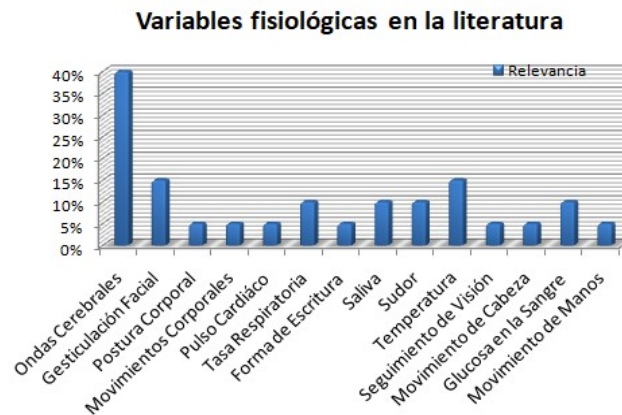


Fig. 1. Relevancia de las variables fisiológicas relacionadas con el nivel de atención en la literatura.

## 2.2 Diadema de MindWave

Para realizar la implementación del sistema electrónico en este trabajo, utilizamos el dispositivo MindWave desarrollado por el fabricante Neurosky ([www.neurosky.com](http://www.neurosky.com)), que permite obtener señales de EEG a través de una interfaz de tipo diadema que se coloca en la cabeza y es alimentado por una batería tipo AAA de 1.5 V. Utiliza una interfaz inalámbrica para comunicarse con la computadora y adquiere las señales a través de sensores biológicos pasivos conectados a un electrodo que hace contacto con la frente. Además, tiene un terminal de referencia conectado al lóbulo de la oreja. Esta característica se usa para determinar el origen de una señal. En las interfaces cerebro-computadora, la ubicación de los electrodos permite obtener diferentes representaciones del EEG. El dispositivo MindWave tiene solo un terminal colocado en la parte frontal del sujeto, en lo que se conoce formalmente como zona pre-frontal. La figura 2 muestra la forma del dispositivo.



Fig. 2. Diadema Mindwave del fabricante Neurosky

### 2.3 Banda para medir la postura corporal

Los resultados de los experimentos basados en las ondas cerebrales del usuario se consideran una variable dependiente y procedemos a relacionar el nivel de atención de una persona que usa la postura corporal como una segunda variable fisiológica.

El sensor se colocó inicialmente en un chaleco para poder realizar la medición (Fig. 4A), esto implicó un tiempo más prolongado en la toma de muestras, por lo que fue reemplazado por una banda como se muestra en la Fig. 4B.

La orientación del sensor MPU-6050 con respecto al sistema tridimensional (X, Y, Z) se puede ver en la Fig. 3. Los ángulos medidos son solo aquellos que forman los movimientos alrededor del eje "X" y movimientos alrededor del eje "Y"; esto se debe a que son los únicos ángulos donde se consideran las reacciones de soporte tónico estático que se producen para mantener una postura normal y vertical contra la fuerza de la gravedad, lo que hace que el usuario requiera un equilibrio neuro-motor en relación con su neuro-eje que produce un impacto directo en su nivel de atención y concentración. Esto es analizado en (Doberssan y Rodríguez, 2004).

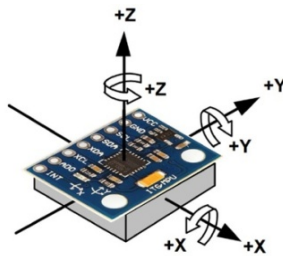


Fig. 3. Referencia de posición del sensor MPU-6050 en la banda.



**Fig. 4** A) Sensor de postura colocado en el chaleco. B) Sensor de postura colocado en la banda del cuerpo.

### 3 Afectación del TDAH en el cerebro

El déficit de atención disperso (cuya abreviatura es DDA y TDAH si la hiperactividad es la más frecuente) es un trastorno de causa poco clara, probablemente con la intervención de factores genéticos y ambientales, en los que existe una alteración a nivel del sistema nervioso central, manifestándose a través de un aumento en la actividad, impulsividad y falta de atención, y frecuentemente asociando otras alteraciones.

El factor genético se demuestra, ya que el TDAH es de 5 a 7 veces más frecuente en los hermanos y de 11 a 18 veces más frecuente en los hermanos gemelos. Se han descrito varios genes posiblemente involucrados.

El TDAH es una de las causas más frecuentes de fracaso escolar y problemas sociales en los niños.

### 4 Desarrollo de la prueba de atención

Las pruebas experimentales se realizaron utilizando el dispositivo comercial MindWave de la compañía Neurosky, para detectar el nivel de atención en estudiantes mexicanos de nivel primario.

Se evaluó una muestra de 22 estudiantes con TDAH diagnosticado y 11 estudiantes sin TDAH cuyas edades están entre 6 y 12 años.

La prueba consistió en una prueba para identificar colores, que se obtuvo de las demostraciones de la compañía Brain HQ (<https://www.brainhq.com/why-brainhq/about-the-brainhq-exercises/attention>).

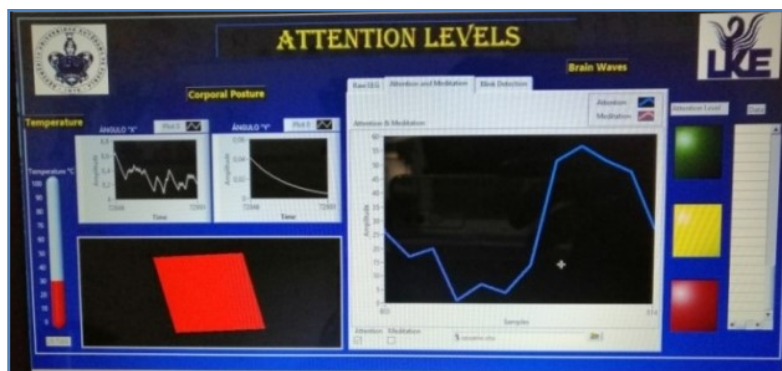
Para obtener los datos de las señales cerebrales, se implementó una interfaz gráfica, utilizando el software LABVIEW. La figura 6 ilustra la interfaz gráfica donde puede observar el comportamiento de las señales cerebrales, la postura del cuerpo, la temperatura, un semáforo como retroalimentación, un vector donde se almacenan los datos muestreados y la variación del nivel de atención del usuario.

La prueba se realizó en la modalidad de tableta para ambos casos (estudiantes con TDAH diagnosticado y estudiantes sin TDAH). El desarrollo práctico se muestra en la Fig. 5.

El experimento se desarrolló en un aula donde el ambiente estaba controlado y era adecuado para evitar agentes que distraen y obtener una respuesta natural y un mejor rendimiento del usuario.



**Fig. 5** Desarrollo de la prueba en estudiantes con TDAH.

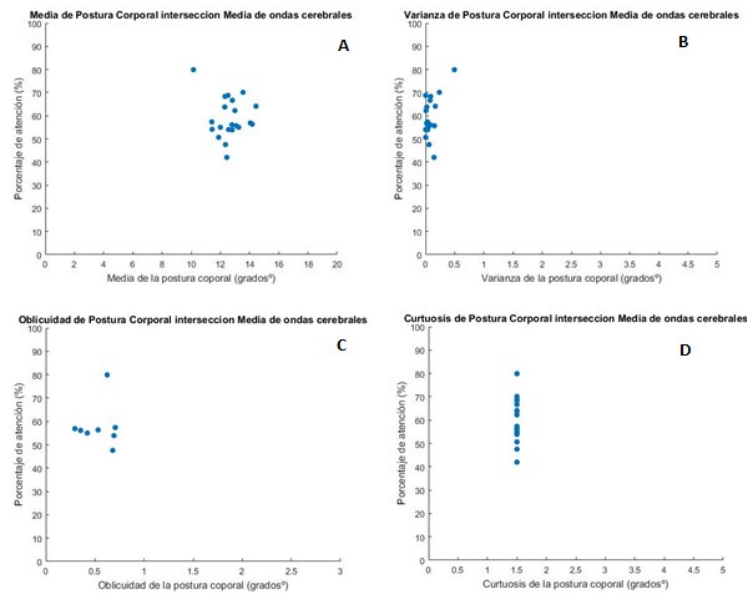


**Fig. 6** Interfaz gráfica implementada en Labview.

## 5 Análisis y resultados

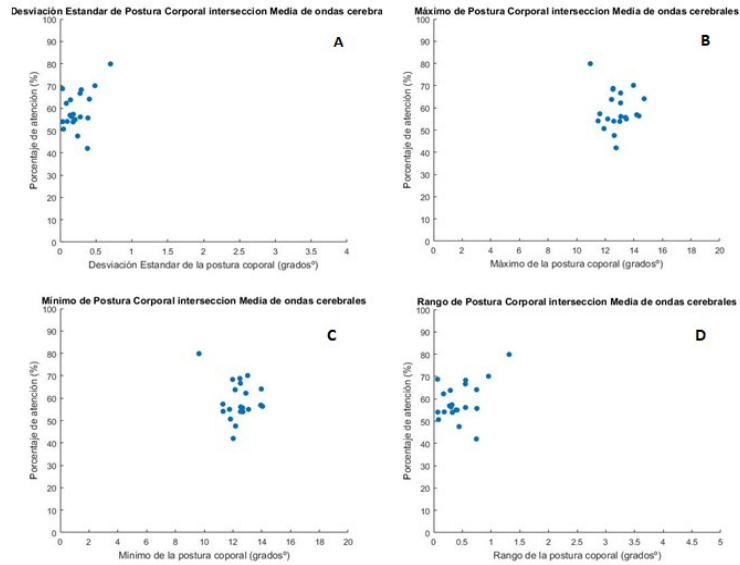
En la visualización de los datos, se trazan diagramas de dispersión 2D, que son una herramienta fundamental para encontrar patrones y tendencias de correlación entre el porcentaje de atención y el resultado de la postura corporal. Los vectores se trazan en la Fig. 7A: (vector PC medio, vector promedio OC), en la Fig. 7B se trazan los vectores (vector de varianza PC, vector promedio OC), en los vectores el vector (vector se oblicuidad PC, promedio OC vector), en la Fig. 7D se trazan los vectores (vector de curtiduría PC, vector promedio OC), en la Fig. 8A se trazan los vectores: (vector de desviación estándar de PC, vector promedio OC), en la Fig. 8B se trazan los vectores (PC vector máximo, vector OC medio), en la Fig. 8C se trazan los vectores (vector PC mínimo, vector OC medio) y en la Fig. 8D se trazan los vectores (PC rango de vectores, vector OC).

Las pruebas experimentales se realizaron utilizando el dispositivo comercial MindWave.



**Fig. 7.** Gráficos de dispersión 2D A) (PC media, OC medio). B) (varianza PC, OC promedio). C) (PC oblicua, media OC). D) (PC curtuosis, medio OC).





**Fig. 8.** Gráficos de dispersión 2D A) (desviación de PC, promedio de OC). B) (PC máximo, OC promedio). C) (PC mínimo, OC promedio). D) (rango de PC, OC promedio).

Utilizando la información analizada, se obtiene el factor de correlación ( $\rho_{xy}$ ) entre cada característica independiente extraída de la postura corporal (X) y la característica dependiente extraída del porcentaje de atención (Y), tanto para niños con TDAH como para niños sin TDAH ( se muestra en la tabla 1).

**Tabla 1.** Coeficientes de correlación por cada característica extraída en niños con TDAH.

Característica Extraída	Coef. De Correlación ( $\rho_{xy}$ ) en niños con TDAH	Coef. De Correlación ( $\rho_{xy}$ ) en niños sin TDAH
Media	0.1737	0.2225
Varianza	0.1098	0.1413
Oblicuidad	0.1059	0.1746
Curtuosis	0.2999	0.1622
Desviación Estandar	0.1628	0.1609
Máximo	0.2183	0.2466
Mínimo	0.2040	0.2678
Rango	0.1611	0.1379

## 6 Conclusiones

Los resultados obtenidos en el análisis de datos muestran que las características estadísticas más influyentes en la correlación entre la postura corporal y el nivel de atención son el mínimo, el máximo y la curtosis.

En los diagramas de dispersión 2D, puede ver los patrones típicos de cada una de las características extraídas de donde puede hacer una clasificación donde se obtienen 2 grupos de membresía o de pertenencia que son: niños con TDAH y niños sin TDAH.

**Acknowledgment:** Reconocimiento especial a la maestra “Claudia González Calleros” por su valiosa colaboración en tomar muestras con estudiantes con TDAH.

## Referencias

- Pascual, M. F., Begoña, Z. & Buldian, K. M. (2010). “Adaptive cognitive rehabilitation interventions based on serious games for children with ADHD using biofeedback techniques: assessment and evaluation”, *COMPUTE '10 Proceedings of the Third Annual ACM Bangalore Conference*, article 29, Bilbao, España, pp. 1-4.
- Asiry, O., Shen, H. & Calder P. (2015). “Extending Attention Span of ADHD Children through an Eye Tracker Directed Adaptive User Interface”, *ASWEC '15 Vol. II Proceedings of the ASWEC 2015 24th Australasian Software Engineering Conference*, volumen 1, Australia, pp. 149- 152.
- Weisberg, O., Galoz, A., Berkowitz, R., Weiss, N., Peretz, O., Azoulai, S., Rubin, D. K. & Zuckerman, O. (2014) “TangiPlan: designing an assistive technology to enhance executive functioning among children with ADHD”, *IDC '14 Proceedings of the 2014 conference on Interaction design and children*, volumen 1, New York, USA, pp. 293-296.
- Sonne & Jensen, M. M. (2016) “Evaluating the ChillFish Biofeedback Game with Children with ADHD”, *IDC '16 Proceedings of the The 15th International Conference on Interaction Design and Children*, volumen 1, New York, USA, pp. 529 – 534.
- Marín, E. J. (2014) “Detección de emociones del usuario”, en Tesis Pontificia Universidad Católica de Valparaíso, volumen 1, Chile, pp. 1- 67.
- Saneiro, M. M. (2015) “Apoyo psico-educativo y afectivo en entornos virtuales de aprendizaje”, *International Journal of Developmental and Educational Psychology*, volumen 1, numero 2, España, De INFAD Base de datos, pp. 233-241.
- García, A. E. (2015) “Análisis de ondas cerebrales para determinar emociones a partir de estímulos visuales”, *Universidad Veracruzana Facultad de Estadística e Informática*, volumen 1, Xalapa, Veracruz, México, 2015, pp. 1-137.
- Dorbessan, L. & Rodríguez, C. A. (2004) “La postura corporal en el deporte simétrico y asimétrico”, *Argentina: En Tesis Universidad Abierta Interamericana*, volumen 1, pp. 1- 15.

# Capítulo 10

## Protocolo para Realización de Experimentos de Elicitación de Gestos Relacionados a Tareas Abstractas

David Céspedes-Hernández, Juan Manuel González-Calleros

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación  
{dcespedesh, jumagoca78}@gmail.com

**Resumen.** El análisis de la comunicación humana en sus distintas modalidades, desde la perspectiva de la IHC, ha motivado la creación de interfaces naturales de usuario. Estas interfaces buscan imitar procesos apoyándose en la tecnología existente de manera que los usuarios puedan interactuar con sistemas de la misma manera en que lo harían con otras personas. El presente trabajo de investigación tiene como objetivo proponer un protocolo para llevar a cabo experimentos de elicitación de gestos considerando tareas a realizar para la preparación de instrumentos, configuración del experimento, el procesamiento de la información obtenida, y su representación. El protocolo de experimento presentado en este artículo incluye la selección de instrumentos de recolección de información, considera la utilización la técnica del Mago de Oz para la conducción del experimento y hace uso de mecanismos para la representación de los gestos obtenidos.

**Palabras clave:** Interacción Gestual, Interacción Humano Computadora, Interfaces Naturales de Usuario, Elicitación de Gestos.

### 1 Introducción

Dentro de las Ciencias Computacionales, la Interacción Humano Computadora (IHC) se encarga de investigar mecanismos que permiten a usuarios interactuar con sistemas de manera eficiente. La inspiración para el desarrollo de tales mecanismos de interacción ha llegado desde diferentes lugares: basándose en tecnología existente, siguiendo principios que han sido propuestos y validados siguiendo el método científico, y más recientemente, considerando la forma en la que las personas se comunican de manera natural.

El análisis de la comunicación humana en sus distintas modalidades, desde la perspectiva de la IHC, ha motivado la creación de interfaces de usuario llamadas naturales (NUI por sus siglas en inglés). En los estudios relacionados a las NUI, tras abstraer características del intercambio de información entre personas, se busca imitar procesos apoyándose en la tecnología existente de manera que los usuarios puedan interactuar con sistemas de

la misma manera en que lo harían con otras personas (i.e. mediante comandos vocales o a través de lenguaje corporal).

La comunicación efectuada a través de movimientos corporales, también llamados gestos, es especialmente interesante, ya que es posible encontrar trabajos cuyos autores mencionan que el 65 por ciento de la comunicación humana se lleva a cabo por medios no verbales (Anders, 2015, p. 123) y, sin embargo, la mayoría de los trabajos relacionados con el procesamiento del lenguaje natural se centran en el análisis de texto y voz. Una revisión preliminar a la literatura arrojó que los dominios de aplicación de las interfaces gestuales son diversos. Existen esfuerzos de parte de autores enfocados en educación, interacción niño-computadora, rehabilitación, entrenamiento atlético, entrenamiento en lenguaje de señas, y autenticación de usuarios (Jia, Hu, Lu & Yuan, 2007, p.60) (Liang & Ouhyoung, 1998, p. 558) (Schlömer, Poppinga, Henze & Boll, 2008, p. 11).

Los problemas a los que se enfrentan los autores y desarrolladores de propuestas relacionadas a la interacción gestual desde un punto de vista técnico se relacionan principalmente con oclusiones que pueden existir entre los usuarios y los dispositivos con los que sus movimientos son monitoreados. La resolución de tales inconvenientes se logra a través de la adición de dispositivos adicionales organizados de forma que unos permitan la visualización de los puntos ciegos de los otros. De esta manera, algunas propuestas manifiestan alcanzar precisión cercana al 100% en el reconocimiento de gestos y una eficacia cercana al tiempo real para tal fin (Kim, Han, Kim, Lee, & Kim, 2017, p. 2975) (Zou, Xiao, Han, Wu, Li, & Ni, 2017, p. 381) (Liu, Sacks, Zhang, Richardson, Lucas & Van der Spiegel, 2017, p. 1257).

Así, actualmente el reto en el reconocimiento gestual no es de carácter técnico sino semántico y relacionado con líneas de investigación de la IHC, ya que los gestos son específicos a culturas y lenguajes, y por lo tanto son ambiguos e incompletamente especificados (Chandler & Schwarz, 2009). Dicho de otra manera, si bien las técnicas y algoritmos propuestos por investigadores han demostrado ser suficientes para la detección y el reconocimiento gestual, no se encontraron en la literatura trabajos en los que se mencione haber identificado vocabularios de gestos que puedan ser asociados a tareas de distintas aplicaciones independientemente de su contexto de uso.

El presente trabajo de investigación tiene como objetivo proponer un protocolo para llevar a cabo experimentos de elicitación de gestos considerando tareas a realizar para la preparación de instrumentos, configuración del experimento, el procesamiento de la información obtenida, y su representación. Esto permitirá asociar los gestos con tareas abstractas para que posteriormente puedan ser computacionalmente identificados, reconocidos y relacionados con actividades específicas en múltiples contextos.

El resto del artículo se organiza en secciones. En la sección dos se reporta el estado del arte describiendo trabajos sobre la elicitación de gestos para distintos propósitos, introduciendo un modelo descriptivo de los movimientos realizables por los usuarios, describiendo tareas estándar o abstractas, presentando la técnica de Mago de Oz para la conducción del experimento, y mostrando un instrumento de evaluación de usabilidad. La tercera

sección se dedica a la presentación de la propuesta del trabajo de investigación incluyendo la definición de pasos y materiales recomendados para el fin expuesto. Finalmente, en la sección 4 se brindan las conclusiones del trabajo realizado y se introduce el trabajo futuro identificado.

## **2 Estado del Arte**

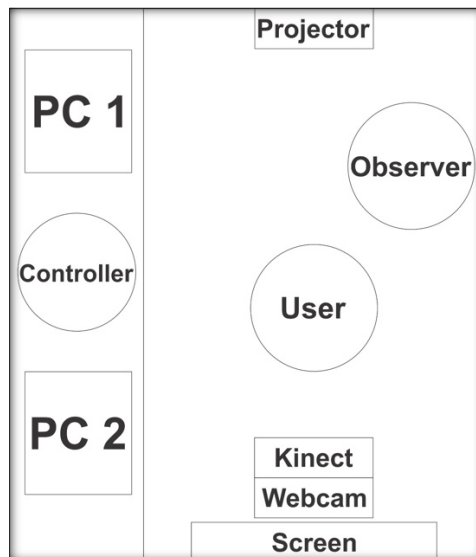
Con el fin de comprender la pertinencia y el aporte de una investigación, es necesario conocer tanto los antecedentes de esta, como los conceptos fundamentales en los que se basa. Es por esto, que en las siguientes subsecciones se presentan trabajos relacionados acerca de la elicitación de gestos para distintos propósitos, se introduce la técnica de Mago de Oz (Höysniemi, Hämäläinen & Turkki, 2004, p. 27) en la que la propuesta se apoya, se aborda la descripción del modelo de capacidad propuesto por Kaklanis (2013, p. 403) como base para la representación de gestos obtenidos, se menciona la propuesta de Constantine (2003, p. 1) sobre la existencia de tareas estándar o abstractas, y se destaca el instrumento CSUQ para evaluación de usabilidad (Lewis, 1995, p. 57).

### **2.1 Estudios de Elicitación de Gestos y la Técnica del Mago de Oz**

En la literatura es posible encontrar trabajos con el objetivo de identificar gestos para diversos dominios. Por ejemplo, Vatavu & Zaiti (2014, p. 131) llevaron a cabo un estudio de elicitación de gestos para controlar una Smart TV. El método utilizado consistió en determinar 21 tareas a ser realizadas interactuando solo con una mano a través de un sensor leap motion. En el experimento participaron 18 personas caracterizadas de acuerdo con su edad, sexo y mano hábil. Los sujetos fueron colocados en un escenario similar al de una sala de televisión convencional, se les mostró el funcionamiento del dispositivo de recolección de gestos y, para cada una de las actividades requeridas, se les recomendaron gestos. Posteriormente, tras un período de preparación, cada uno de los usuarios efectuó movimientos a manera de propuesta. Finalmente, cada uno de los sujetos evaluó los gestos realizados utilizando una escala de Likert de 5 puntos. Un observador del experimento registró los tiempos que tomaba a los usuarios efectuar los gestos tras haber recibido la instrucción. Como resultados de este trabajo, sus autores refieren haber encontrado un conjunto de gestos equivalentes a comandos que pueden ser ejecutados durante la interacción con Smart TVs. Así mismo, en el artículo se reportan tiempos totales de ejecución de gestos, valores de acuerdo entre los múltiples usuarios, calificaciones brindadas por los usuarios a los gestos realizados, y se mencionan restricciones del trabajo.

En la actualidad, un dominio recurrente en el que se aplica la interacción gestual es el automotriz. Tal es el caso del trabajo de May, Gable & Walker (2017, p. 74), donde se lleva a cabo un estudio de elicitación de gestos para ejecutar una serie de acciones

ubicando al usuario en un simulador que representa el ambiente en el que se encontraría al conducir un automóvil. En el experimento participaron 14 sujetos caracterizados de acuerdo con su sexo y edad. En total, se evaluaron los gestos para seis tareas relacionadas a la navegación en listas o menús. Tras la realización del experimento, se reportaron los gestos identificados junto con las puntuaciones de acuerdo correspondientes a cada uno de ellos. Para llevar a cabo el experimento reportado, se utilizó el ‘enfoque de teatro’, en el que se le presentan a los sujetos estímulos ilustrando el resultado de la interacción y posteriormente se les pide que realicen el gesto que creen que podría haber dado tal resultado.



**Fig. 1.** Configuración de experimento basado en la técnica del Mago de Oz.

La técnica del Mago de Oz (Höysniemi, et al., 2004, p. 27), consiste en diseñar una experiencia similar a la que un usuario tendría al interactuar con un sistema sin necesidad de llegar a la implementación de esta. A través de la utilización de recursos multimedia y prototipos de interfaz, es posible así llevar a cabo experimentos que permitan observar cómo se comportaría el usuario ante ciertas tareas que se espera que sea capaz de realizar con el sistema a desarrollar y así, identificar errores en etapas tempranas de desarrollo.

La técnica del Mago de Oz ha sido utilizada a lo largo de la historia del desarrollo de sistemas interactivos, y en particular, en el campo del desarrollo de las NUI como una forma para recolectar datos en ambientes de realidad mixta (Dow, Lee, Oezbek, MacIntyre, Bolter, Gandy, 2005) y para encontrar comandos gestuales para interacción entre otros dominios de aplicación. La selección de esta estrategia para la presente

propuesta se basa en la intención de evitar el desarrollo de sistemas complejos y enfatizar el esfuerzo en el análisis gestual sobre la implementación de medios de reconocimiento.

De la mano con el conocimiento sobre cómo se pueden identificar gestos para interacción, es necesario comprender qué tipo de movimientos pueden ser realizados por el usuario, es decir, cuáles son los puntos y articulaciones que deben ser monitoreados durante el experimento. Para este fin, en la siguiente subsección se describe un modelo de usuario haciendo énfasis en las capacidades humanas.

## 2.2 Modelo de Capacidad

Para los propósitos de este artículo, y alineado con la definición del problema, esta subsección se enfoca en la descripción de un método para modelar los movimientos que los usuarios potencialmente pueden realizar.

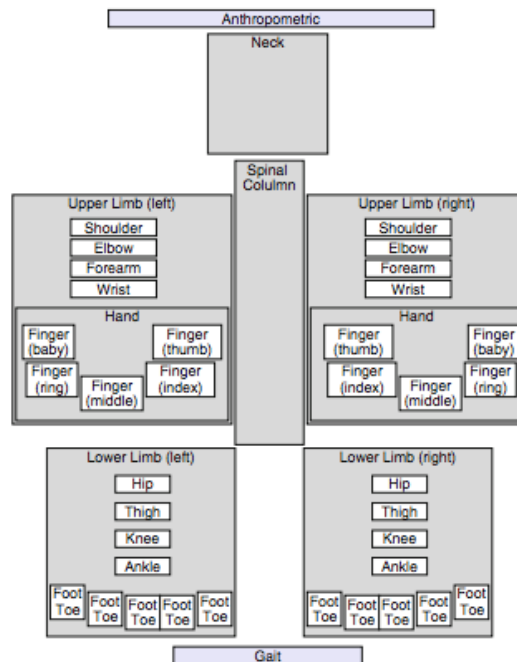


Fig. 2. Modelo virtual de usuario (GVUM) de Kaklanis (2013, p. 403).

De entre las propuestas al respecto, para el presente trabajo de investigación se toma en cuenta la propuesta de Kaklanis (2013, p. 403), en la que se detalla un modelo de usuario virtual (GVUM) y se asocia con un modelo de capacidades. De acuerdo con lo que se

puede observar en el modelo GVUM presentado en la Fig. 2., para llevar a cabo el análisis de capacidades, el cuerpo humano se forma de seis componentes principales: cuello, columna vertebral, extremidad superior izquierda, extremidad superior derecha, extremidad inferior izquierda, y extremidad inferior derecha. En un nivel más detallado, las extremidades superiores incluyen hombros, codos, antebrazos, muñecas, y manos con dedos, mientras que las extremidades inferiores incluyen a su vez la cadera, muslos, rodillas, tobillos y dedos del pie.

Al utilizar el GVUM, es posible modelar a los usuarios de manera anatómica, pero no lo es representar su interacción con las interfaces de un sistema. Para satisfacer este objetivo, el modelo se acompaña de un modelo de capacidades que consta de 108 clases correspondientes a sistemas, partes del cuerpo, características y movimientos realizables tales como rotaciones, flexiones, extensiones, adducciones y abducciones por mencionar algunos. La capacidad para identificar los movimientos que los usuarios pueden realizar al interactuar con sistemas permite llevar a cabo experimentos en los que se identifiquen los gestos que estos usan con mayor frecuencia y su emparejamiento con tareas específicas.

Una vez que se ha definido la forma en la que se puede obtener información sobre los gestos, es necesario identificar las tareas para las que tales gestos funcionarán. En la siguiente subsección se describen tareas estándar que pueden servir como base para el planteamiento de estudios sobre interacción gestual.

### **2.3 Tareas Estándar**

Con el objetivo de modelar sistemas informáticos, en (Constantine, 2003, p. 1) se propusieron las tareas estándar que se llevan a cabo en los mismos, junto con los componentes y prototipos abstractos que les dan soporte. Estas tareas estándar, determinadas por verbos, son acciones que se efectúan típicamente y que, al agregarles sustantivos, definen tareas específicas de las soluciones informáticas creadas. Es importante resaltar que las tareas corresponden a funciones interactivas y no a componentes de interfaz.

Las tareas abstractas identificadas son las siguientes: operar, ir a, iniciar, detener, completar, seleccionar, crear, borrar, eliminar, modificar, mover, duplicar, ejecutar, alternar y ver. Definir las instrucciones a realizar en términos de estas tareas estándar, proporciona la oportunidad de generalizar los resultados tras la culminación del experimento y predecir los gestos que pueden ser utilizados para otras concretizaciones de ellas. Es decir, a partir de un estudio de elicitación para un contexto específico, generar un vocabulario de gestos que puede ser empleado en otro contexto.

A partir de la definición de las tareas a analizar y tomando en cuenta los movimientos a observar, se puede llevar a cabo el experimento en el que se realiza el emparejamiento entre tareas y gestos. La siguiente sección está dedicada a la descripción del protocolo para tal fin.



### 3 Protocolo para la Realización de Experimentos de Elicitación de Gestos Relacionados a Tareas Abstractas

El protocolo propuesto en el presente trabajo consta de tres etapas: la preparación de los instrumentos para la obtención de gestos, la configuración del experimento, y el procesamiento de la información que se obtiene tras la realización del experimento. Las siguientes subsecciones describen cada una de tales etapas.

#### 3.1 Preparación de Instrumentos para Reporte de un Experimento de Elicitación de Gestos

La presente propuesta consta de tres etapas. En la primera etapa, se debe obtener el consentimiento de parte de los usuarios para hacer uso de la información que proporcionan y se debe hacer de acuerdo con lineamientos éticos y legales. Igualmente, en esta etapa preparatoria, se debe llevar a cabo la caracterización de los usuarios conforme a la información que para el experimento en curso resulte útil y la definición de las tareas que se observarán durante el experimento y para ese fin se propone hacer uso de un formato que, además de detallar las instrucciones a llevar a cabo, es de utilidad para registrar los resultados del experimento. La estructura del instrumento propuesto para enlistar las tareas a llevar a cabo y para posteriormente añadir los resultados del estudio durante la segunda etapa, se presenta en la Tabla 1.

**Table 1.** Instrumento propuesto para registro de tareas e incorporación de resultados preliminares.

Identificador de tarea	Nombre de la tarea	Tarea estándar relacionada	Identificador del gesto	Tiempo	Calificación

Una vez que el experimento ha sido conducido y que se han obtenido los datos de este tanto en los formatos de registro de gestos, tiempos y calificaciones, como en videos y notas de parte del observador y del controlador del experimento, como parte de esta propuesta se considera solicitar a los sujetos que tras su intervención en el experimento respondan el cuestionario CSUQ (Lewis, 1995, p. 57) con el fin de tener datos que revelen información adicional acerca de su percepción de la usabilidad del sistema simulado. Finalmente, se debe llevar a cabo el procesamiento de la información reunida.

El procesamiento de la información obtenida tras la realización de experimentos de elicitación se deriva a la presentación de resultados. Además de los formatos con tareas y detalles de la interacción, y de los resultados de promedios y desviaciones estándar reportados por el instrumento CSUQ sobre las dimensiones de usabilidad, es necesario presentar el vocabulario de los gestos recolectados.

Para la presentación del conjunto de gestos, se puede optar por representaciones textuales informales (e.g. levantamiento del brazo derecho), representaciones textuales de los movimientos de acuerdo con notación como la incluida en el modelo GVUM presentado en la sección 2.2 (e.g. flexión del hombro derecho, extensión del codo derecho), por medio de representaciones gráficas o animaciones, o formalmente mediante el uso de estructuras de datos que permitan la representación de los gestos desde un punto de vista computacional. Como parte de la presente propuesta, se creó una estructura conforme a lo detallado en la sección 2.2 conteniendo tanto partes del cuerpo y articulaciones como los movimientos y características propias de ellos. En la Fig. 3. se muestra un extracto del archivo generado.

```

257 <leftLowerLimb>
258   <hip>
259     <flexion></flexion>
260     <extension></extension>
261     <adduction></adduction>
262     <abduction></abduction>
263     <externalRotation></externalRotation>
264     <internalRotation></internalRotation>
265   </hip>
266   <knee>
267     <flexion></flexion>
268     <extension></extension>
269   </knee>
270   <ankle>
271     <dorsiFlexion></dorsiFlexion>
272     <plantarFlexion></plantarFlexion>
273     <eversión></eversión>
274     <inversion></inversion>
275     <pronation></pronation>
276     <supination></supination>
277     <lateralRotation></lateralRotation>
278     <medialRotation></medialRotation>
279   </ankle>
280   <foot>
281     <footToe1>
282       <flexion></flexion>
283       <extension></extension>
284     </footToe1>
285     <footToe2>
286       <flexion></flexion>
287       <extension></extension>
288     </footToe2>
289     <footToe3>
290       <flexion></flexion>
291       <extension></extension>
292     </footToe3>

```

**Fig. 3.** Extracto de la estructura generada para presentación de los gestos obtenidos.

Tras la preparación de los instrumentos para recolección de datos, se puede llevar a cabo el experimento. La siguiente subsección se dedica a reportar la configuración de este.

### 3.2 Configuración del Experimento

La configuración del experimento siguiendo la técnica del Mago de Oz presentada en la sección 2.1 y mostrada en la Fig.1., incluye dos computadoras (PC1 y PC2), la primera para controlar una cámara de profundidad, en este caso un sensor Microsoft Kinect©, y una cámara para obtener información de los movimientos de los usuarios y por lo tanto,

para permitir la documentación del experimento; mientras que la segunda se conecta a un proyector para simular la interacción del usuario con el sistema y para brindarle retroalimentación. Previo a la conducción del experimento con los usuarios, es necesario solicitarles que brinden información de acuerdo con lo descrito en la subsección 3.1 sobre documentos de consentimiento informado, descripción demográfica y cuestionarios de apreciación de las NUI según sea necesario para los objetivos específicos del experimento.

Durante el experimento, se plantea que un observador colocado junto con el usuario en el área de experimentación, le solicite ejecutar gestos para llevar a cabo las tareas de interés. El observador puede además brindar soporte en caso de que el usuario presente dudas o tenga algún tipo de inconveniente durante la realización de las actividades. El controlador del experimento debe asegurarse de mantener la simulación y de recolectar datos que podrán ser reforzados en etapas posteriores de procesamiento, tales como el tiempo que le toma al usuario realizar los gestos tras recibir las instrucciones, las calificaciones que el usuario le brinda a cada uno de los gestos que realiza, y expresiones observadas en el usuario como aparente frustración, ansiedad, confusión entre otras que pudieran condicionar los resultados obtenidos.

Tras la realización del experimento, la presente propuesta comprende que se le solicite al usuario completar los instrumentos de evaluación con respecto a la usabilidad del sistema. Posteriormente, es necesario integrar la información obtenida, realizar cálculos y generalizaciones, y presentar los hallazgos realizados de la manera en que se explicó en la subsección anterior calculando niveles de acuerdo para gesto, destacando los gestos que obtuvieron las mejores puntuaciones y caracterizándolos de manera que puedan ser replicados en otros contextos. Las conclusiones al presente trabajo y el trabajo futuro identificado se presenta en la siguiente sección.

## **4 Conclusiones y Trabajo Futuro**

En el presente artículo se reporta el diseño de un protocolo para llevar a cabo estudios de elicitación de gestos considerando la técnica del Mago de Oz y la alineación de los gestos obtenidos con tareas estándar.

Parte de la propuesta incluye la descripción de una tabla en la que se recomienda registrar las tareas que formarán parte del experimento a conducir, relacionándolas con tareas abstractas y brindando la oportunidad de indicar en ella, además, los tiempos de ejecución de cada gesto y la calificación que el usuario que los realiza les da. Finalmente, se menciona la creación de una estructura formal basada en el modelo GVUM descrito en la sección 2.2, en el que se pueden caracterizar los gestos con la intención de que puedan ser posteriormente replicados fielmente.

Como trabajo futuro, se considera la elaboración de avatares 3D que, utilizando la información en los archivos generados, puedan representar gráficamente los gestos relacio-

nados con determinadas tareas. Así mismo, se plantea agregar al protocolo de elicitación de gestos una etapa adicional en la que los vocabularios de gestos sean evaluados por usuarios ajenos al experimento de elicitación, con el fin de validar la interacción en distintos contextos utilizando tales conjuntos de gestos. Finalmente, se plantea contemplar como parte del protocolo la posibilidad de utilizar el ‘enfoque de teatro’ mencionado en la sección 2.1 con el fin de contrastar los resultados obtenidos a través de la utilización de ambas técnicas para elicitación.

## Referencias

- Anders, S. (2015), “What are you really saying?” *Physician leadership journal*, Vol. 2(2), pp. 82-83.
- Chandler, J., y Schwarz, N. (2009), “How extending your middle finger affects your perception of others: Learned movements influence concept accessibility”. *Journal of Experimental Social Psychology*, Vol. 45(1), pp. 123-128.
- Constantine, L. L. (2003), “Canonical abstract prototypes for abstract visual and interaction design”. *International Workshop on Design, Specification, and Verification of Interactive Systems*, pp. 1-15.
- Dow, S., Lee, J., Oezbek, C., MacIntyre, B., Bolter, J. D., y Gandy, M. (2005), “Wizard of Oz Interfaces for Mixed Reality Applications”, *CHI 2005, ACM*. <https://doi.org/10.1145/1056808.1056911>.
- Höysniemi, J., Hämäläinen, P., y Turkki, L. (2004). “Wizard of Oz prototyping of computer vision-based action games for children”. *Conference on Interaction design and children: building a community*, pp. 27-34.
- Jia, P., Hu, H. H., Lu, T., y Yuan, K. (2007), “Head gesture recognition for hands-free control of an intelligent wheelchair”. *Industrial Robot*, Vol. 34(1), pp. 60-68.
- Kaklanis, N., Moschonas, P., Moustakas, K., y Tzovaras, D. (2013). “Virtual user models for the elderly and disabled for automatic simulated accessibility and ergonomics evaluation of designs”. *Universal access in the information society*, Vol. 12(4), pp. 403-425.
- Kim, S. Y., Han, H. G., Kim, J. W., Lee, S., y Kim, T. W. (2017). “A hand gesture recognition sensor using reflected impulses”. *IEEE Sensors Journal*, Vol. 17(10), pp. 2975-2976.
- Lewis, J.R. (1995), “IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use”, *International Journal of Human-Computer Interaction*, Vol. 7, No. 1, pp. 57-78. <https://doi.org/10.1080/10447319509526110>.
- Liang, R. H., y Ouhyoung, M. (1998), “A real-time continuous gesture recognition system for sign language”. *Third IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 558-567.
- Liu, X., Sacks, J., Zhang, M., Richardson, A. G., Lucas, T. H., y Van der Spiegel, J. (2017). “The virtual trackpad: An electromyography-based, wireless, real-time, low-power, embedded hand-gesture-recognition system using an event-driven artificial neural network”. *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 64, pp. 1257-1261.

- May, K. R., Gable, T. M., y Walker, B. N. (2017). "Designing an in-vehicle air gesture set using elicitation methods". *9th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pp. 74-83.
- Schlömer, T., Poppinga, B., Henze, N., y Boll, S. (2008), "Gesture recognition with a Wii controller". *2nd international conference on Tangible and embedded interaction*, pp. 11-14.
- Vatavu, R. D., y Zaiti, I. A. (2014). "Leap gestures for TV: insights from an elicitation study". *ACM International Conference on Interactive Experiences for TV and Online Video*, pp. 131-138.
- Zou, Y., Xiao, J., Han, J., Wu, K., Li, Y., y Ni, L. M. (2017). "Grfid: A device-free rfid-based gesture recognition system". *IEEE Transactions on Mobile Computing*, Vol. 16(2), pp. 381-393.

# Capítulo 11

## Estudio de la cinemática de un robot de 18 grados de libertad

Miguel Angel Ortega Palacios<sup>1</sup>, Amparo Dora Palomino Merino<sup>2</sup>, José Fernando Reyes Cortés<sup>2</sup>

Benemérita Universidad Autónoma de Puebla

<sup>1</sup>Facultad de Ciencias de la Computación, <sup>2</sup>Facultad de Ciencias de la Electrónica

miguel.ortegap@alumno.buap.mx, ampalomino@gmail.com

fernando.reyes@correo.buap.mx

**Resumen.** Emular el proceso de marcha en un robot bípedo presenta numerosos retos. Hoy en día hay esfuerzos significativos en esta dirección, pero son más los ejemplos de fallas en los mecanismos de marcha que hacen vigente la búsqueda de alternativas de solución. La propuesta de investigación usa como caso de estudio el robot Bioloid de 18 grados de libertad. El objetivo del presente trabajo es el estudio de su cinemática, por lo que se hace una revisión del estado del arte sobre el estudio de este robot para validar los alcances y limitaciones que se pueden tener al momento de implementar la marcha. Además, se aspira a que la investigación aquí desarrollada pueda ser utilizada para el desarrollo de nuevos métodos de implementación de marcha en robots bípedos.

**Palabras Clave:** Marcha, robot bípedo, cinemática.

### 1 Introducción

A medida que más robots móviles comiencen a integrarse en diferentes áreas de la sociedad, necesitarán operar en una amplia variedad de entornos, incluso en aquellos que pueden ser peligrosos para el ser humano. El empleo de robots bípedos autónomos es cada vez más frecuente, se pueden utilizar para diversas aplicaciones, tales como: navegación, búsqueda, cooperación, vigilancia, y transporte. Pero el desarrollo de este tipo de robots sigue siendo un gran desafío para la comunidad científica, por lo tanto, es fundamental encontrar métodos más robustos para la ejecución de la marcha en robots bípedos.

El resto de este documento se estructura de la siguiente forma: en la sección dos se presenta el estudio del estado del arte, en la tercera sección se presenta el modelo cinemático directo del robot Bioloid, en la cuarta sección la simulación de la marcha, y finalmente en la quinta sección las conclusiones.

## 2 Estado del arte

Se hizo un estudio de los trabajos que han empleado el robot Bioloid en la literatura para validar los alcances y limitaciones que se pueden tener al implementar la marcha en el robot. Por lo tanto, el presente estudio está enfocado los métodos y técnicas empleadas para implementar la marcha en este robot. En la Tabla 1 se muestra un análisis con respecto a las técnicas empleadas para implementar la marcha o el movimiento de algunas articulaciones del robot Bioloid, de esta manera se pudo observar los elementos principales que se necesitan para lograr modelar e implementar la marcha en este robot.

**Tabla 1.** Técnicas y métodos empleados para modelar la marcha del robot Bioloid.

Artículo	Generación de rutinas	Modelo cinemático	Modelo dinámico	Punto de momento Zero (ZMP)	Redes neuronales	Ondas sinusoidales acopladas	Algoritmo genético	Aprendizaje por refuerzo
(Han, Ha, & Kim, 2008)	■							
(Meggiolaro, Neto, & Figueroa, 2016)		■					■	
(Segura, García, Díaz, & Pech, 2017)	■							
(Nunez, Briseno, Rodriguez, Ibarra, & Rodriguez, 2012)		■						
(Reyes, Pamanes, Fierro, & Nunez, 2018)		■		■				
(Krishnan, Aswath, & Udupa, 2014)	■							
(Bravo, Rengifo, & Cortes, 2015)				■				■
(Lin, Hwang, Jiang, & Chen, 2016)		■						
(Cerritos-Jasso, y otros, 2013)		■						
(Arias, Olvera, Pámanes, & Núñez, 2014)		■						
(Hein, Hild, & Berger, 2007)					■			
(Eaton, 2007)						■	■	
(Wolf, Hall, Robinson, &	■							

Culverhouse, 2007)			
(Teodoro, 2007)			
(Akhtaruzzaman & Shafie, 2011)			
(Pfeiffer, 2011)			
(Núñez, Sapiens, Rodríguez, Briceño, & Rodríguez, 2011)			
(Brenes Torres, 2017)			
(Gonzalez, Mage, & Rodas, 2015)			
(Bravo Montenegro, 2016)			

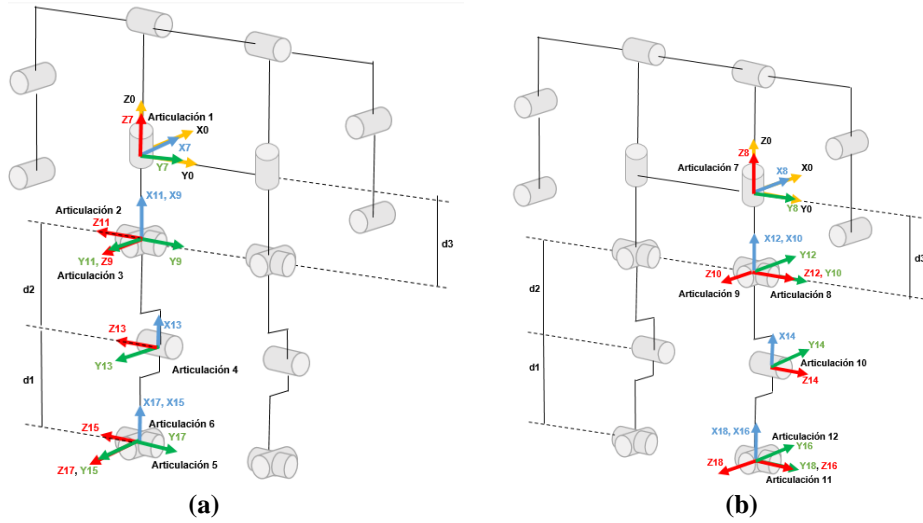
### 3 Modelo Cinemático directo

Para la descripción de la cinemática del robot Bioloid, se asignaron marcos de referencia ortonormales aplicando la convención modificada de Denavit-Hartenberg (Londoño Palacio, Maldonado Granados, & Calderón Villafañez, 2014).

Empleando la convención de Denavit-Hartenberg se propone establecer cuatro marcos de referencia fijos para formar cuatro cadenas cinemáticas abiertas: la primera corresponde a la pierna derecha (Figura 1 (a)), la segunda cadena cinemática pertenece a la pierna izquierda (Figura 1 (b)), la tercera al brazo derecho (Figura 2 (a)), y la última al brazo izquierdo (Figura 2 (b)).

Para la primera cadena cinemática, los marcos se enumeran de manera consecutiva desde la articulación uno (eslabón fijo asignado a la cadera derecha), hasta la seis (eslabón final asignado al tobillo derecho), tal como se aprecia en la Figura 1 (a). Para la segunda cadena cinemática, los marcos se enumeran de manera consecutiva desde la articulación siete (asignado al eslabón fijo que corresponde a la cadera izquierda), hasta la doce (eslabón final asignado al tobillo izquierdo), esto se puede ver en la Figura 1 (b). Es importante mencionar, que cada servomotor del robot tiene un identificador asignado, Los servomotores que van del ID1 al ID6 corresponden a las articulaciones superiores, es decir al movimiento de los brazos izquierdo y derecho, mientras que los servomotores que van del ID7 al ID18 corresponden a las articulaciones inferiores, es decir las piernas izquierda y derecha.





**Figura 1.** (a) Marcos de referencia asociados a la cadena cinemática de la pierna derecha del robot Bioloid. (b) Marcos de referencia asociados a la cadena cinemática de la pierna izquierda del robot Bioloid.

En la Tabla 2 se proporcionan los parámetros de Denavit-Hartenberg para la cadena cinemática que corresponde a la pierna derecha del robot. Donde:  $\theta_a = \theta_7 + \frac{\pi}{4}$ .

**Tabla 2.** Parámetros de Denavit-Hartenberg para la cadena cinemática de la pierna derecha del robot Bioloid Premium.

Articulación	Servomotor	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
1	ID 7	$0^\circ$	0	$\theta_a$	0
2	ID 1	$90^\circ$	0	$\theta_{11}$	0
3	ID 9	$90^\circ$	0	$\theta_9$	0
4	ID 13	$-90^\circ$	$d_2$	$\theta_{13}$	0
5	ID 15	$0^\circ$	$d_1$	$\theta_{15}$	0
6	ID 17	$90^\circ$	0	$\theta_{17}$	0

Posteriormente se hace el mismo análisis para la cadena cinemática del brazo y la pierna izquierda del robot.

Para el modelo utilizado se definieron las siguientes distancias:  $d_1 = d_2 = 7.5$  cm.,  $d_3 = 3.4$  cm,  $d_4 = 6.8$  cm y  $d_5 = 1.7$  cm.

(Aguilar & Vivas, 2009) definen la matriz de transformación general para articulaciones arborescentes como:

$$T_i = \begin{pmatrix} \cos(\theta_i) & -\text{sen}(\theta_i) & 0 & d_i \\ \cos(\alpha_i)\text{sen}(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\text{sen}(\alpha_i) & -r_i\text{sen}(\alpha_i) \\ \text{sen}(\alpha_i)\text{sen}(\theta_i) & \text{sen}(\alpha_i)\cos(\theta_i) & \cos(\alpha_i) & r_i\text{sen}(\alpha_i) \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1)$$

Utilizando la ecuación (1) y aplicando los términos definidos en la Tabla 2 se obtienen las siguientes matrices de transformación para cada articulación:

$$T_1 = \begin{pmatrix} \cos(\theta_a) & -\text{sen}(\theta_a) & 0 & 0 \\ \text{sen}(\theta_a) & \cos(\theta_a) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_2 = \begin{pmatrix} \cos(\theta_{11}) & -\text{sen}(\theta_{11}) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \text{sen}(\theta_{11}) & \cos(\theta_{11}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_3 = \begin{pmatrix} \cos(\theta_9) & -\text{sen}(\theta_9) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \text{sen}(\theta_9) & \cos(\theta_9) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_4 = \begin{pmatrix} \cos(\theta_{13}) & -\text{sen}(\theta_{13}) & 0 & d_2 \\ 0 & 0 & 1 & 0 \\ -\text{sen}(\theta_{13}) & -\cos(\theta_{13}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_5 = \begin{pmatrix} \cos(\theta_{15}) & -\text{sen}(\theta_{15}) & 0 & d_1 \\ \text{sen}(\theta_{15}) & \cos(\theta_{15}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_6 = \begin{pmatrix} \cos(\theta_{17}) & -\text{sen}(\theta_{17}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \text{sen}(\theta_{17}) & \cos(\theta_{17}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

En la Tabla 3 se proporcionan los parámetros de Denavit-Hartenberg para la cadena cinemática que corresponde a la pierna izquierda del robot. Donde:  $\theta_b = \theta_8 - \frac{\pi}{4}$ .

**Tabla 3.** Parámetros de Denavit-Hartenberg para la cadena cinemática de la pierna izquierda del robot Bioloid Premium.

Articulación	Servomotor	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
7	ID 8	0°	0	$\theta_b$	0
8	ID 12	90°	0	$\theta_{12}$	0
9	ID 10	-90°	0	$\theta_{10}$	0
10	ID 14	90°	d2	$\theta_{14}$	0
11	ID 16	0°	d1	$\theta_{16}$	0
12	ID 18	-90°	0	$\theta_{18}$	0

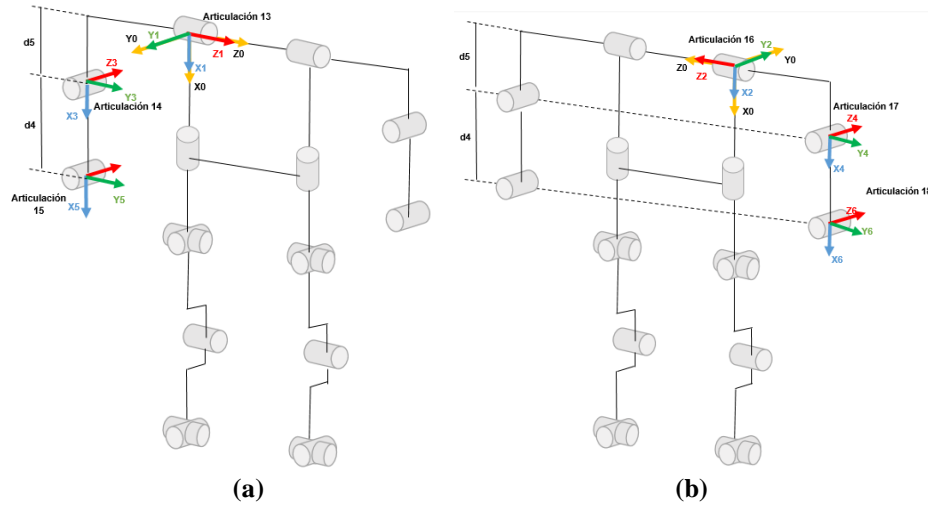
Utilizando la ecuación (1) y aplicando los términos definidos en la Tabla 3 se obtienen las siguientes matrices de transformación para cada articulación:

$$T_7 = \begin{pmatrix} \cos(\theta_b) & -\text{sen}(\theta_b) & 0 & 0 \\ \text{sen}(\theta_b) & \cos(\theta_b) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_8 = \begin{pmatrix} \cos(\theta_{12}) & -\text{sen}(\theta_{12}) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \text{sen}(\theta_{12}) & \cos(\theta_{12}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_9 = \begin{pmatrix} \cos(\theta_{10}) & -\text{sen}(\theta_{10}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\text{sen}(\theta_{10}) & -\cos(\theta_{10}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_{10} = \begin{pmatrix} \cos(\theta_{14}) & -\text{sen}(\theta_{14}) & 0 & d_2 \\ 0 & 0 & -1 & 0 \\ \text{sen}(\theta_{14}) & \cos(\theta_{14}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_{11} = \begin{pmatrix} \cos(\theta_{16}) & -\text{sen}(\theta_{16}) & 0 & d_1 \\ \text{sen}(\theta_{16}) & \cos(\theta_{16}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_{12} = \begin{pmatrix} \cos(\theta_{18}) & -\text{sen}(\theta_{18}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\text{sen}(\theta_{18}) & -\cos(\theta_{18}) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

En la tercera cadena cinemática, los marcos se enumeran de manera consecutiva desde la articulación trece (asignado al eslabón fijo que corresponde al hombro derecho), hasta la quince (eslabón final asignado a la mano derecha), como se muestra en la Figura 2 (a). Por último, en la cuarta cadena cinemática, los marcos se enumeran de manera consecutiva desde la articulación dieciséis (asignado al eslabón fijo que corresponde al hombro izquierdo), hasta el dieciocho (eslabón final asignado a la mano izquierda), como se aprecia en la Figura 2 (b).



**Figura 2. (a)** Marcos de referencia asociados a la cadena cinemática del brazo derecho del robot Bioloid. **(b)** Marcos de referencia asociados a la cadena cinemática del brazo izquierdo del robot Bioloid.

La Tabla 4 proporciona los parámetros de Denavit-Hartenberg para la cadena cinemática que corresponde al brazo derecho del robot. Donde:  $\theta_c = \theta_1 + \frac{\pi}{2}$  y  $\theta_d = \theta_3 - \frac{\pi}{2}$ .

**Tabla 4.** Parámetros de Denavit-Hartenberg para la cadena cinemática del brazo derecho del robot Bioloid Premium.

Articulación	Servomotor	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
13	ID 1	$0^\circ$	0	$\theta_c$	0
14	ID 3	$-90^\circ$	d5	$\theta_d$	0
15	ID 5	$0^\circ$	d4	$\theta_5$	0

Utilizando la ecuación (1) y aplicando los términos definidos en la Tabla 4 se obtienen las siguientes matrices de transformación para cada articulación:

$$T_{13} = \begin{pmatrix} \cos(\theta_c) & -\text{sen}(\theta_c) & 0 & 0 \\ \text{sen}(\theta_c) & \cos(\theta_c) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_{14} = \begin{pmatrix} \cos(\theta_d) & -\text{sen}(\theta_d) & 0 & d_5 \\ 0 & 0 & 1 & 0 \\ -\text{sen}(\theta_d) & -\cos(\theta_d) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_{15} = \begin{pmatrix} \cos(\theta_5) & -\text{sen}(\theta_5) & 0 & d_4 \\ \text{sen}(\theta_5) & \cos(\theta_5) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La Tabla 5 proporciona los parámetros de Denavit-Hartenberg para la cadena cinemática que corresponde al brazo izquierdo del robot. Donde:  $\theta_e = \theta_2 - \frac{\pi}{2}$  y  $\theta_f = \theta_4 + \frac{\pi}{2}$ .

**Tabla 5.** Parámetros de Denavit-Hartenberg para la cadena cinemática del brazo izquierdo del robot Bioloid Premium.

Articulación	Servomotor	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
16	ID 2	$0^\circ$	0	$\theta_e$	0
17	ID 4	$90^\circ$	d5	$\theta_f$	0
18	ID 6	$0^\circ$	d4	$\theta_6$	0

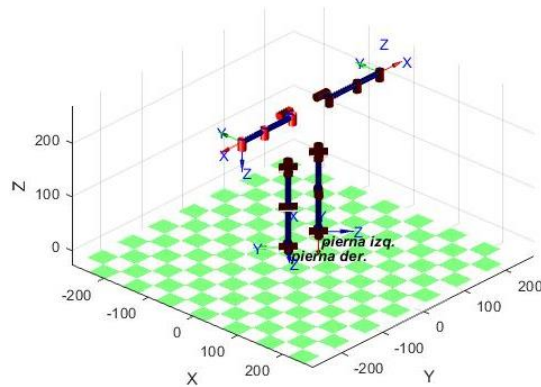
Utilizando la ecuación (1) y aplicando los términos definidos en la Tabla 5 se obtienen las siguientes matrices de transformación para cada articulación:

$$T_{16} = \begin{pmatrix} \cos(\theta_e) & -\text{sen}(\theta_e) & 0 & 0 \\ \text{sen}(\theta_e) & \cos(\theta_e) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_{17} = \begin{pmatrix} \cos(\theta_f) & -\text{sen}(\theta_f) & 0 & d_5 \\ 0 & 0 & -1 & 0 \\ \text{sen}(\theta_f) & \cos(\theta_f) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_{18} = \begin{pmatrix} \cos(\theta_6) & -\text{sen}(\theta_6) & 0 & d_4 \\ \text{sen}(\theta_6) & \cos(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

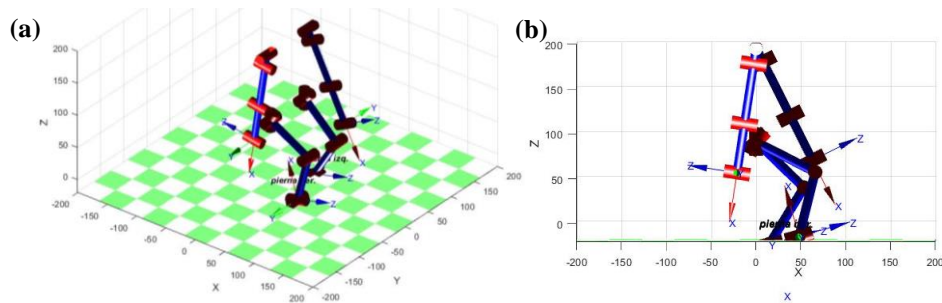
## 4 Simulación de marcha

Posteriormente se realizó la simulación de la marcha utilizando el toolbox de robótica de Matlab. Primero se creó el modelo del robot, empleando los parámetros de Denavit-Hartenverg obtenidos en la sección anterior. El modelo resultante se observa en la Figura 3. En el cual las posiciones articulares de todos los eslabones están en cero.

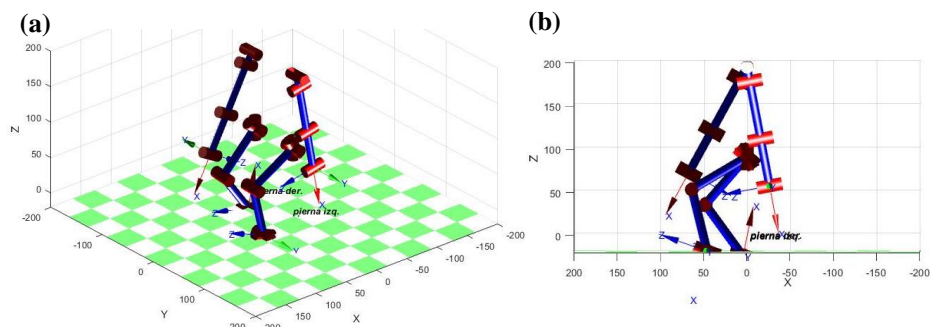


**Figura 3.** Posición de las articulaciones cuando los servomotores están en la posición angular de  $0^\circ$ .

El siguiente paso fue asignar las posiciones articulares de cada uno de los servomotores para que el robot se capaz de ejecutar el movimiento de marcha, para lograr esto, se tomaron los datos de las posiciones articulares del software RoboPlus motion. De esta manera, al ejecutar el mismo ciclo de movimientos de manera repetida, se logra simular la marcha del robot. En la Figura 4 se muestra la posición de los brazos y piernas cuando el robot da un paso hacia adelante con el pie derecho. Mientras que en la Figura 5 se muestra la posición de los brazos y piernas en el momento que el robot da un paso al frente con el pie izquierdo.



**Figura 4.** Posición de las articulaciones cuando el robot mueve la pierna derecha al frente: (a) Vista isométrica. (b) Vista lateral.



**Figura 5.** Posición de las articulaciones cuando el robot mueve la pierna izquierda al frente. **(a)** Vista isométrica. **(b)** Vista lateral.

## 5 Conclusiones

El análisis del estado del arte, la mayor parte de los trabajos consideran el estudio del modelo cinemático del robot, por lo que es un paso importante para modelar la marcha en un robot bípedo. En la simulación solo consideró el modelo cinemático directo, empleando los datos de las posiciones articulares que tiene el software del robot, estos resultados preliminares servirán para el desarrollo de estrategias para simular el movimiento de la marcha sin tener que utilizar la base de datos que tiene el software, para que posteriormente pueda ser implementado en el robot. Por ahora no se considera la dinámica del robot, por lo tanto, el equilibrio del robot durante la marcha dependerá solamente de que los movimientos de cada una de las articulaciones no sean tan bruscos para que el robot pueda mantenerse de pie en cada movimiento ejecutado. Además, no se considera la orientación y traslación de las articulaciones del robot.

## Referencias

- Aguilar, D., & Vivas, A. (2009). Modelado y control de un robot bípedo de nueve grados de libertad. *In VIII Congreso de la Asociación Colombiana de Automática.*
- Akhtaruzzaman, M., & Shafie, A. A. (2011). Geometrical analysis on BILOID humanoid system standing on single leg. *In 2011 4th International Conference on Mechatronics (ICOM)*, 1-5.
- Arias, L. E., Olvera, L. I., Pámanes, J. A., & Núñez, J. V. (2014). Patrón de marcha 3D de tipo cicloidal para humanoides y su aplicación al robot Bioloid. *Revista Iberoamericana de Ingeniería Mecánica.*
- Bravo Montenegro, D. A. (2016). Generación de Trayectorias para un Robot Bípedo basadas en Captura de Movimiento Humano (Doctoral dissertation).

- Bravo, D. A., Rengifo, C. F., & Cortes, A. (2015). Simuladores en robótica: Qué validar, la cinemática o la dinámica? Caso de estudio: Simulación de postura de un robot bípedo.
- Brenes Torres, J. C. (2017). API de generación de trayectorias para robot Bioloid sobre Beaglebone Black.
- Cerritos-Jasso, J. R., Camarillo-Gómez, K. A., Monsiváis-Medina, J. A., Castillo-Alfaro, G., Pérez-Soto, G. I., & Pámanes-García, J. A. (2013). Kinematic Modeling of a Humanoid Soccer-Player: Applied to BIOLOID Premium Type A Robot. *In FIRA RoboWorld Congress*, 49-63.
- Eaton, M. (2007). Evolutionary humanoid robotics: past, present and future. *In 50 years of artificial intelligence*, 42-52.
- Gonzalez, J. T., Mage, H. N., & Rodas, C. R. (2015). Simulación del Ciclo de Marcha del Robot Bípedo Bioloid en el Entorno Virtual V-REP. *In de Congreso Mexicano de Robotica*.
- Han, J., Ha, I., & Kim, B. (2008). Educational robotic construction kit: Bioloid. *Proceedings of 17th World Congress of the International Federation*, 6-10.
- Hein, D., Hild, M., & Berger, R. (2007). Evolution of biped walking using neural oscillators and physical simulation. *In Robot Soccer World Cup*. 433-440.
- Krishnan, A. B., Aswath, S., & Udupa, G. (2014). Real Time Vision Based Soccer Playing Humanoid Robotic Platform. *In Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing*, 52.
- Lin, J. L., Hwang, K. S., Jiang, W. C., & Chen, Y. J. (2016). Gait balance and acceleration of a biped robot based on Q-learning. 2439-2449.
- Londoño Palacio, O. L., Maldonado Granados, L. F., & Calderón Villafañez, L. C. (2014). Guías para construir estados del arte.
- Meggiolaro, M. A., Neto, M. S., & Figueroa, A. L. (2016). Modeling and optimization with genetic algorithms of quasi-static gait patterns in planar biped robots.
- Nunez, J. V., Briseno, A., Rodriguez, D. A., Ibarra, J. M., & Rodriguez, V. M. (2012). Explicit analytic solution for inverse kinematics of bioloid humanoid robot. *Brazilian Robotics Symposium and Latin American Robotics Symposium*, 33.
- Núñez, V., Sapiens, J., Rodríguez, D., Briceño, A., & Rodríguez, V. (2011). Interface para interacción con robots humanoids. *In Actas del XIII Congreso Mexicano de Robótica (COMROB)*.
- Pfeiffer, S. (2011). Guiado gestual de un robot humanoide mediante un sensor" kinect".
- Reyes, G., Pamanes, J. A., Fierro, J. E., & Nunez, V. (2018). Optimum Walking of the Bioloid Humanoid Robot on a Rectilinear Path. *Computational Kinematics*, 143-151.
- Segura, C. G., García, M. G., Díaz, L. N., & Pech, R. B. (2017). Robot Bioloid Premium jaranero controlado remotamente por voz.
- Teodoro, P. (2007). Humanoid robot: Development of a simulation environment of an entertainment humanoid robot.
- Wolf, J., Hall, P., Robinson, P., & Culverhouse, P. (2007). Bioloid based humanoid soccer robot design. *In the Proc. of the Second Workshop on Humanoid Soccer Robots*.

# Capítulo 12

## Búsqueda informada de coeficientes de la ecuación de movimiento de un cuadricópteros mediante redes neuronales convolutivas

Carlos Leopoldo Carreón Díaz de León<sup>1</sup>, Sergio Vergara Limon<sup>2</sup>, M. Aurora D. Vargas Treviño<sup>2</sup>

<sup>1</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

<sup>2</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Electrónica

dermandad@hotmail.com

**Resumen.** En este trabajo se propone una hipótesis para la búsqueda de los coeficientes de la ecuación de movimiento de un cuadricóptero utilizando redes neuronales convolutivas (CNN por sus siglas en inglés). La importancia de hallar estos coeficientes radica en la capacidad de diseñar e implementar aeronaves con un consumo eficiente de energía y con un sistema computacional adecuado para el control del movimiento. Se estudia brevemente el marco conceptual relacionado al tema junto con un breve análisis del estado del arte. Se plantea una hipótesis junto a los objetivos y una justificación de la importancia que conlleva la realización de este trabajo.

**Palabras Clave:** Cuadricópteros, Modelo dinámico, Red neuronal convolutiva.

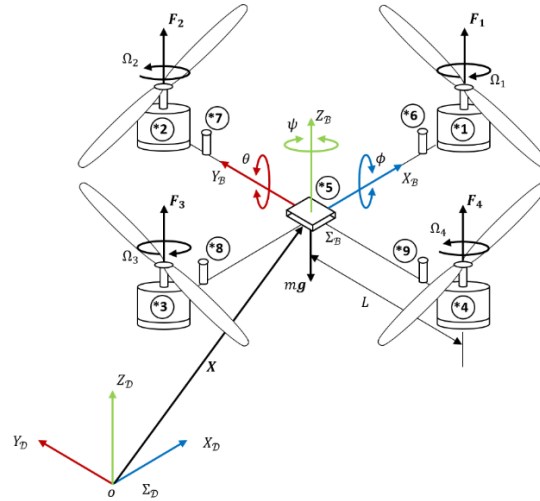
### 1 Introducción

Los cuadricópteros son máquinas voladoras que generan un vector de fuerza de sustentación controlado por medio de la fricción aerodinámica de sus actuadores Castillo Pedro (2009). Su construcción simétricamente distribuida permite su utilización en aplicaciones de transporte, vigilancia por adquisición de imagen Omid Ghorbanzadeh (2019) entre otras.

El comportamiento del movimiento del cuadricóptero es complejo ya que se trata de un sistema que es sub-actuado; las variables de entrada que son cuatro deben controlar seis variables de salida: tres de orientación y tres de posición. La ecuación de movimiento (EM) o modelo dinámico describe cómo se comporta el cuadricóptero con respecto al tiempo considerando los coeficientes de la estructura mecánica de la aeronave. En la figura 1 se



observa un esquema simplificado de un cuadricóptero y en la figura 2 se muestra un cuadricóptero real.



**Fig. 1.** Esquema de un cuadricóptero.  $\Sigma_D$  es el marco de referencia fijo con sus respectivos ejes cartesianos,  $\Sigma_B$  es el marco de referencia móvil con sus ejes móviles,  $[x_i, y_i, z_i]^T \forall i \in \{D, B\}$  son los ejes cartesianos para cada sistema de referencia,  $[\phi, \theta, \psi]^T$  son los ángulos de orientación.  $\Omega_j \forall j \in \{1,2,3,4\}$  son las velocidades de los rotores que son actuados por los cuatro motores numerados del 1 al 4 mientras que del número 6 a 9 son los sensores de velocidad. El N° 5 es el centro geométrico del cuadricóptero.



**Fig. 2.** Cuadricóptero construido en la Facultad de Ciencias de la Electrónica de la BUAP.

La EM del cuadricóptero es:

$$\frac{d}{dt} \mathbf{x}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\varphi}, t) \in \mathbb{R}^{12} \quad (1)$$

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}, \boldsymbol{\varphi}, t) \in \mathbb{R}^6$$

donde

- $\mathbf{x}(t)$ : Vector de estados de la EM.
- $\mathbf{y}(t)$ : Señales observables de la EM.
- $\mathbf{u}(t)$ : Vector de entradas de la EM.
- $\boldsymbol{\varphi}$ : Coeficientes de la EM.
- $\mathbf{f}, \mathbf{g}$ : Función de estados y de salida respectivamente.

El tipo de estructura de  $\mathbf{f}$  y  $\mathbf{g}$  son características de cada robot o aeronave (*ver* Carlos L. (2019) *para la estructura del cuadricóptero*). El vector de estados  $\mathbf{x}(t)$  incluye las posiciones, velocidades y aceleraciones que interactúan en el cuadricóptero mientras que las señales observables  $\mathbf{y}(t)$  son las que son medidas directamente con sensores o transductores. Las entradas  $\mathbf{u}(t)$  son las entradas de energía para producir el movimiento deseado.

Las funciones  $\mathbf{f}$  y  $\mathbf{g}$  están relacionadas con los coeficientes  $\boldsymbol{\varphi}$  ya que en conjunto describen completamente el movimiento del cuadricóptero independientemente del vector de estados, de las señales de entrada y de las salidas observables. Esto implica que el conocimiento de estas partes de la EM implica el conocimiento completo del sistema, además implica que se puede predecir el movimiento Jonch R. Taylor (2005).

El control del movimiento del cuadricóptero utiliza estructuras matemáticas para suministrar en forma controlada la energía  $\mathbf{u}(t)$  en función de  $\mathbf{y}(t)$  y del movimiento deseado  $\mathbf{y}_d(t)$ . En implementaciones reales del control de movimiento el vector  $\mathbf{y}(t)$  no puede ser medido completamente al igual que la energía  $\mathbf{u}(t)$  no suministra todo el dominio de los números reales, en ecuaciones esto es representado por (2):

$$\mathbf{y}'(k) \subseteq \mathbf{y}(t) \in \mathbb{R}^6 \quad (2)$$

$$\mathbf{u}'(k) \subseteq \mathbf{u}(t) \in \mathbb{R}^4$$

El control de movimiento es  $\mathbf{u}'(k) = C(\mathbf{y}', \mathbf{y}'_d, k)$  donde  $C$  es la ecuación de control. Para obtener un control de movimiento eficiente energéticamente, el control de movimiento queda definido como:

$$\mathbf{u}'(k) = C(\mathbf{y}', \mathbf{x}', \mathbf{y}'_d, \mathbf{f}, \mathbf{g}, \boldsymbol{\varphi}, k) \quad (3)$$

En la ecuación (3) se observa una aplicación al conocimiento de los coeficientes de movimiento y de la estructura de la EM. Este artículo está organizado en la siguiente forma: en la sección 2 se plantea el problema a resolver para hallar los coeficientes de la EM utilizando las señales observables  $\mathbf{y}'(k)$ ,  $\mathbf{u}'(k)$  y la estructura de la EM  $\mathbf{f}, \mathbf{g}$ . En la sección 3 se presenta la hipótesis, la justificación y los objetivos. En la sección 4 se presenta un breve análisis del estado del arte de la inteligencia artificial y los cuadricópteros.

## 2 Planteamiento del problema

La búsqueda de los coeficientes de la EM es un problema que requiere del conocimiento de la estructura de EM, de las mediciones de las señales observables y de las entradas. Considerando los sistemas digitales utilizados para el control de movimiento de los cuadricópteros, es necesario diseñar un algoritmo de búsqueda que identifique cuando los conjuntos de señales  $\mathbf{y}'(k)$  y  $\mathbf{u}'(k)$  convergen a los coeficientes  $\boldsymbol{\varphi}$  considerando las funciones  $\mathbf{f}$  y  $\mathbf{g}$ .

Las variables independientes de este problema son  $\mathbf{y}'(k)$  y  $\mathbf{u}'(k)$  mientras que las variables dependientes son los coeficientes  $\boldsymbol{\varphi}$  considerando que se conoce las funciones  $\mathbf{f}$  y  $\mathbf{g}$ . El algoritmo de búsqueda es informado ya que se sabe cómo debería visualizarse la EM por lo que se necesita de un algoritmo que identifique cuando el conjunto  $\{\mathbf{y}'(k), \mathbf{u}'(k), \boldsymbol{\varphi}\}$  converge a la estructura de la EM.

Para poder identificar la estructura de la EM es necesario crear nueva información a partir de la que se observa para que sea utilizada por el algoritmo que identifica la estructura de la EM. Existen diversas técnicas que extraen más información a partir de un dominio menor.

La transformada de Fourier crea un dominio que utiliza el doble de números para representar la misma información que es introducida Peter V. O'Neil (2018). En la figura 3 se observa una señal de audio que en el dominio de la frecuencia se obtiene más información que solo analizando la señal original.

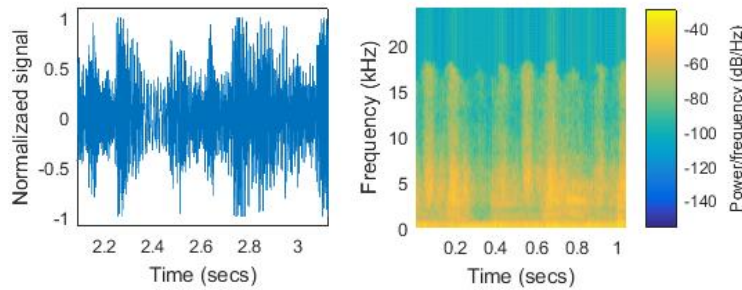
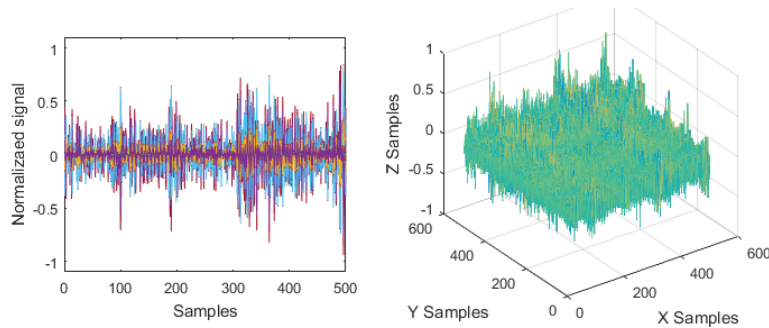


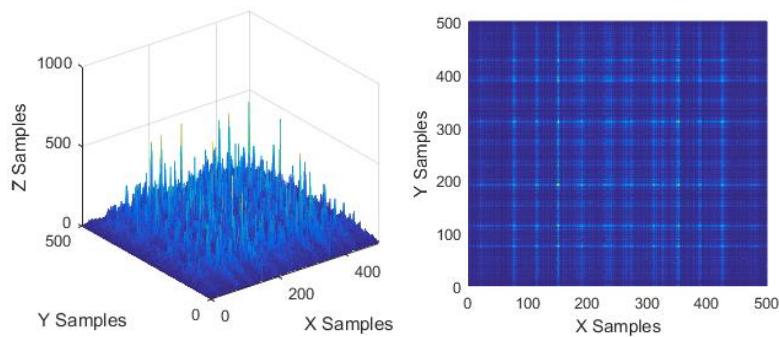
Fig. 3. Transformada rápida de Fourier aplicada a una señal de audio.

Utilizando una matriz de correlación entre los dos canales de la señal de audio, se obtiene otra representación de la información de entrada mostrada en la figura 4:



**Fig. 4.** Matriz de correlación de los dos canales de la señal de audio.

Al aplicar la transformada de Fourier en dos dimensiones, se obtiene más información sobre la señal de audio:



**Fig. 5.** Transformada de Fourier de la matriz de correlación de los dos canales de la señal de audio.

Las técnicas mostradas son aplicables para el conjunto de señales  $\mathbf{y}'(k)$  y  $\mathbf{u}'(k)$  por lo que generando una imagen, se pueden aplicar técnicas de reconocimiento pero aplicadas a identificar la estructura de la EM. Con la identificación de la estructura de la EM se tiene un sistema que permite definir cuando los coeficientes de la EM han sido encontrados y el error de clasificación converge a un mínimo fijado.

Las redes neuronales convolutivas ofrecen un gran desempeño para clasificar imágenes colocando un conjunto de ejemplos de entrada y las salidas deseadas para cada ejemplo Fernando Berzal (2018). La estructura de una red neuronal convolutiva (CNN) está basada en filtros convolutivos en las capas convolutivas, funciones no lineales de activación, reducción de tamaño y terminan en una red neuronal completamente conectada que clasifica las imágenes de entrada como se muestra en la figura 6.

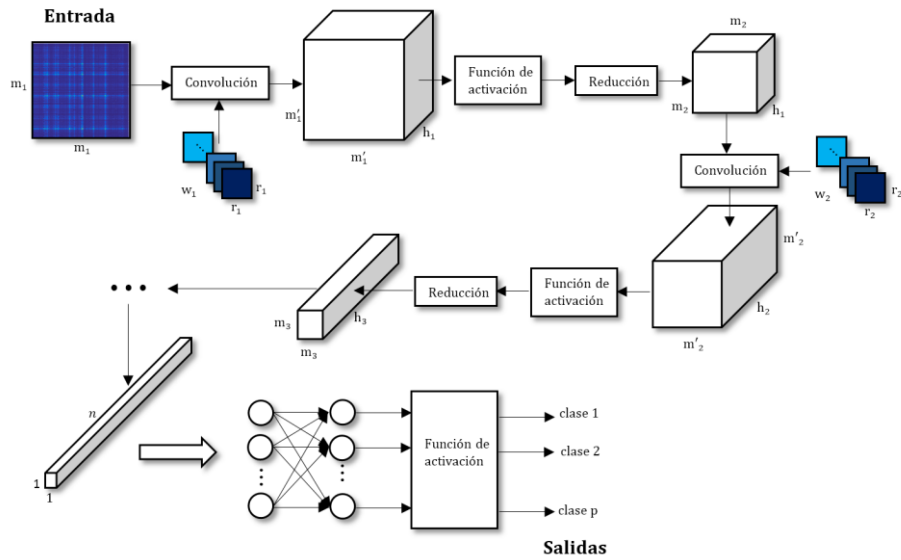


Fig. 6. Esquema de una CNN como clasificador.

El algoritmo de búsqueda informada debe realizar las siguientes tareas: determinar cuando el conjunto de señales de entrada y de salida muestran una imagen que corresponde con la estructura de la EM y a partir de ello utilizar una función heurística que permita converger a los coeficientes de la EM en un tiempo polinómico. En la figura 7 se muestra el algoritmo de búsqueda que se propone.

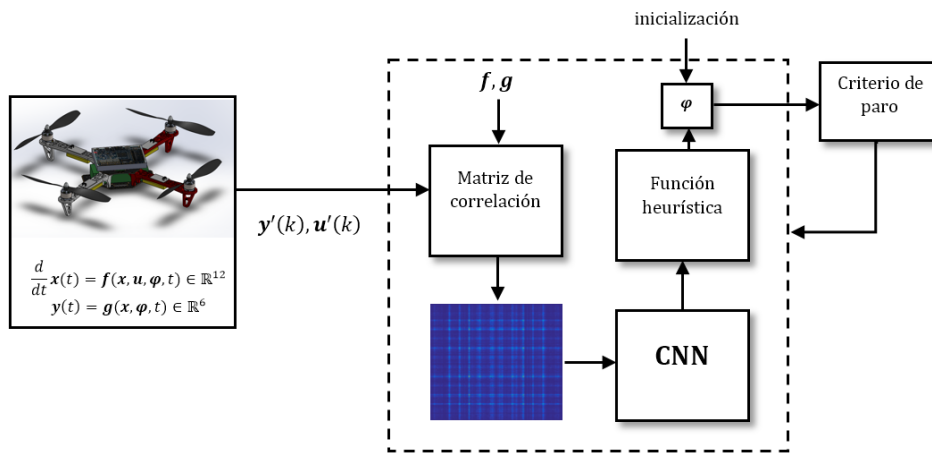


Fig. 7. Algoritmo de búsqueda de coeficientes. La línea punteada indica que es un proceso iterativo.

### **3 Planteamiento de la hipótesis**

#### **3.1. Hipótesis**

La búsqueda de los coeficientes de la EM de un cuadricóptero es realizada considerando un conjunto de datos observables de entrada que al ser procesador por una red neuronal se decide si los coeficientes conforman la estructura de la EM previamente entrenada en la red neuronal. Las salidas de la red neuronal son utilizadas por una función heurística que permite ajustar los coeficientes hasta alcanzar la convergencia hacia los coeficientes reales de la EM del cuadricóptero utilizado en un tiempo polinómico. La red neuronal tiene la capacidad de responder adecuadamente ante incertidumbres de la EM, de la medición de las señales y de la variación de los mismos coeficientes ante cambios en el ambiente.

#### **3.2. Objetivos**

- 1) Diseñar una matriz de correlación con las señales observables y las entradas del sistema considerando el conocimiento de la EM para obtener una representación clara de la estructura de la EM.
- 2) Diseñar una red neuronal convolutiva para identificar cuando la matriz de correlación representa la estructura de la EM y esta coincide con la EM entrenada del cuadricóptero.
- 3) Diseñar una función heurística que reduzca el espacio de búsqueda de los coeficientes de la EM considerando las salidas de la red neuronal convolutiva y un criterio de paro para salir del algoritmo iterativo.
- 4) Implementar el procesamiento del algoritmo de búsqueda informada en un sistema embebido FPGA (*Filed Programmable Gate Array*) para su utilización en la experimentación real con un cuadricóptero.

#### **3.3. Objetivo general**

Diseñar e implementar un algoritmo de búsqueda informada de los coeficientes de la EM de un cuadricóptero utilizando una red neuronal convolutiva y una función heurística de búsqueda.

#### **3.4. Justificación**

La EM permite conocer con precisión el movimiento del cuadricóptero por lo que el diseño de la aeronave se conduce hacia mejorar la eficiencia energética, mecánica y computacional del sistema. Además, se amplía las posibilidades de nuevos sistemas de control como el mostrado en la ecuación (3). El proceso de búsqueda de los coeficientes de la EM continúa siendo un desafío ya que las metodologías encontradas no reflejan las propiedades del movimiento del cuadricóptero. El algoritmo propuesto puede interpolarse a varios tipos de robots ya que la mayoría de sistemas están basados en una EM.

## 4 Breve análisis del estado del arte

La implementación del algoritmo de búsqueda propuesto no es trivial ya que en los cuadricópteros, la medición de las señales observables del cuadricóptero representa una línea de investigación. En Xinhua Wang (2018) se propone un estimador de posición cartesiano que utiliza las señales provenientes de un GPS junto con las obtenidas por un conjunto de acelerómetros montados en el cuadricóptero. En Xinlong Jiang (2015) se utiliza la combinación de señales WIFI junto con acelerómetros, giroscopios y magnetómetros medidas de un celular para estimar la posición de una persona en interiores. Se utiliza una red neuronal de aprendizaje profundo que al ser entrenada con una técnica de entrenamiento extremo Yuan Lan (2009), se obtiene la clasificación de las huellas digitales WIFI y la regresión de la distancia que recorre la persona, se alcanza una precisión de 0.15 metros en promedio. En Balaguer Benjamin (2015) se utilizan diversos puntos de acceso WIFI para generar un mapa bidimensional que indica la ubicación del cuadricóptero en el punto donde se mide la potencia de la señal WIFI. Es aplicado a cuadricópteros para navegación autónoma. Se logra una precisión de ubicación de 3.5 metros.

Las redes neuronales son utilizadas para clasificar los distintos tipos de cuadricópteros que vuelan en un espacio controlado Mohammad F. Al-Sa'd (2019). En este trabajo se utiliza el aprendizaje profundo para clasificar las señales de radio frecuencia para identificar accesos no autorizados. En Serge Mghabghab (2018) se utiliza una CNN para personalizar dos controles de entrada que ayudan a controlar el cuadricóptero en interiores con obstáculos. En Christian Koed Pedersen (2018) se utiliza una red neuronal recurrente con plasticidad sináptica que evita obstáculos considerando los sensores de entrada de posición.

Una de las aplicaciones más interesantes es la detección de fallas en los sensores y transductores. En Alireza Abbaspoura (2017), Dingfei Guo (2018), Payam Aboutalebi (2018), Yujiang Zhong (2019), Samir Zeghlache (2016) se utilizan técnicas con redes neuronales que utilizan la EM y técnicas de predicción para detectar fallas en los componentes del cuadricóptero.

La búsqueda de coeficientes es conocida también como identificación paramétrica. En Saderla Subrahmanyam (2018) se identifican los coeficientes aerodinámicos de una aeronave tipo ala delta. Se realiza experimentación con el sistema de vuelo en un túnel de viento. Los resultados muestran una varianza de los parámetros baja en comparación a la magnitud de los coeficientes. La EM estudiada es linealizada por lo que facilita la identificación. Se utiliza un método de mínimos cuadrados en el dominio de la frecuencia.

En Jeremias B. Machado (2020) se identifica el comportamiento de la señal producida por el movimiento de un cuadricóptero en el eje z utilizando un modelo de ecuación denominado Volterra que en conjunto con un conjunto de ecuaciones ortogonales normalizadas se estima el comportamiento del movimiento. Los trabajos relacionados con la identificación paramétrica son guiados en encontrar una regresión de las señales observables del cuadricóptero mediante una función no lineal estimada por redes neuronales, sin embargo, no se han encontrado trabajos hasta el momento que aborden la

estructura de la EM de un cuadricóptero ya que es una tarea complicada por las siguientes razones:

- 1) La medición de las señales del cuadricóptero es una tarea compleja que requiere grandes capacidades de cómputo para obtener una precisión útil en la búsqueda de los coeficientes de la EM.
- 2) La implementación de los algoritmos de inteligencia artificial requiere de arquitecturas digitales como microprocesadores en paralelo que sean energéticamente eficientes pero con gran capacidad de cómputo.
- 3) El conocimiento de la EM con la gran mayoría de fenómenos físicos no es trivial y necesita del estudio de la física involucrada en el cuadricóptero.

## 5 Conclusiones

En este trabajo se ha mostrado el planteamiento de una hipótesis para la búsqueda informada de los coeficientes de la ecuación de movimiento de un cuadricóptero basada en inteligencia artificial. Se ha planteado el problema a resolver con la metodología propuesta, haciendo énfasis en las propiedades de las señales obtenidas de un sistema digital real utilizado en los cuadricópteros. En el breve análisis del estado del arte de la inteligencia artificial relacionada con los cuadricópteros no se han encontrado trabajos relacionados con la búsqueda de los coeficientes de la EM debido a que representa un gran desafío en el desarrollo teórico y en la implementación de un sistema real.

El conocimiento de los coeficientes de la EM permite realizar diseños eficientes de las aeronaves lo que ayuda a disminuir costos de producción ya que se dimensiona adecuadamente los sistemas computacionales utilizados. Esto implica disminuir el consumo de energía y aumentar la autonomía del robot. Además, el conocimiento de la EM ayuda investigar nuevos esquemas de control para los cuadricópteros para controlar el movimiento con más precisión y seguridad.

## Referencias

- Alireza Abbaspoura, Payam Aboutalebic, Kang K.Yena, Arman Sargolzaei. (2017). "Neural adaptive observer-based sensor and actuator fault detection in nonlinear systems: Application in UAV", ISA Transactions, vol. 67, (pp. 317-329).
- Balaguer Benjamin, Gorkem Erinc, Stefano Carpin. (2015). "Real-time WiFi localization of heterogeneous robot teams using an online random forest", Autonomous Robots, vol. 39, (pp. 155-167).
- Carlos L. Carreón Díaz de León. (2019). "Diseño e implementación de un cuadricóptero guiado por control inalámbrico", Benemérita Universidad Autónoma de Puebla.



- Castillo Pedro, Lozano Rogelio, E. Dzul Alejandro. (2009). "Modelling and Control of Mini-Flying Machines", Springer.
- Christian Koed Pedersen, Poramate Manoonpong. (2018). "Neural Control and Synaptic Plasticity for Adaptive Obstacle Avoidance of Autonomous Drones", Lecture Notes in Computer Science, vol. 10994.
- Dingfei Guo, Maiying Zhong, and Donghua Zhou. (2018). "Multisensor Data-Fusion-Based Approach to Airspeed Measurement Fault Detection for Unmanned Aerial Vehicles", IEEE Transactions on Instrumentation and Measurement, vol. 67.
- Fernando Berzal. (2018). "Redes convolutivas", en Fernando Berzal (1) Matemática avanzadas para ingeniería (pp. 599-679), Edición independiente.
- Jeremias B. Machado, Sidney N. Givigi. (2020). "Convolutional Neural Networks as Asymmetric Volterra Models Based on Generalized Orthonormal Basis Functions", IEEE Transactions on Neural Networks and Learning Systems, vol. 31, (pp. 950-959).
- Jonch R. Taylor. (2005). "Ecuaciones de Lagrange", en Jonch R. Taylor (1) Mecánica clásica (pp. 261-296), Reverté.
- Mohammad F. Al-Sa'd, Abdulla Al-Ali, Amr Mohamed, Tamer Khattab Aiman Erbad. (2019). "RF-based drone detection and identification using deep learning approaches: An initiative towards a large open source drone database", Future Generation Computer Systems, vol. 100, (pp. 86-97).
- Omid Ghorbanzadeh, M., Sansar Raj Meena, Thomas Blaschke, Jagannath Aryal. (2019). "UAV-Based Slope Failure Detection Using Deep-Learning Convolutional Neural Networks", Remote Sensing, vol. 11.
- Payam Aboutalebi, Alireza Abbaspour, Parisa Forouzannezhad, Arman Sargolzaei. (2018). "A Novel Sensor Fault Detection in an Unmanned Quadrotor Based on Adaptive Neural Observer", Journal of Intelligent & Robotic Systems, vol. 90, (pp. 473-484).
- Peter V. O'Neil. (2008). "La integral de fourier y las transformadas de fourier", en Peter V. O'Neil (6) Redes Neuronales & Deep learning (pp. 103-163), Cengage.
- Saderla Subrahmanyam, Yoonsoo Kim, Ghosh A.K. (2018). "Online system identification of mini cropped delta UAVs using flight test methods", Aerospace Science and Technology, vol. 80, (pp. 337-353).
- Samir Zeghlache, Djamel Saigaa, Kamel Kara. (2016). "Fault tolerant control based on neural network interval type-2 fuzzy sliding mode controller for octorotor UAV", Frontiers of Computer Science, vol. 10, (pp. 657-672).
- Serge Mghabghab, Imad H. Elhadj, Daniel Asmar. (2018). "Personalized teleoperation via intention recognition", Advanced Robotics: New Horizons in Telerobotics for Real-Life Applications, vol. 32.
- Xinhua Wang, Weicheng Wang. (2018). "Nonlinear Signal-Correction Observer and Application to UAV Navigation", IEEE Transactions on Industrial Electronics, vol. 6, pp. 4600-4607.
- Xinlong Jiang, Yiqiang Chen, Junfa Liu, Dingjun Liu, Yang Gu, Zhenyu Chen. (2015). "Real-Time and Accurate Indoor Localization with Fusion Model of Wi-Fi Fingerprint and Motion Particle Filter", Mathematical Problems in Engineering, vol. 2015.
- Yuan Lan, Yeng Chai Soh, Guang-Bin Huang. (2009). "Ensemble of online sequential extreme learning machine", Neurocomputing, vol. 72 (pp. 3391-3395).
- Yujiang Zhong, Wei Zhang, Youmin Zhang, Junyi Zuo, Hao Zhan. (2019). "Sensor Fault Detection and Diagnosis for an Unmanned Quadrotor Helicopter", Journal of Intelligent & Robotic Systems, vol. 96, (pp. 555-572).

# Capítulo 13

## Implementación de redes neuronales para el cálculo automático de los parámetros de configuración de un robot cartesiano

Daniel Marcelo González Arriaga<sup>1</sup>, María Aurora Diozcora Vargas Treviño<sup>2</sup>, Sergio Vergara Limon<sup>2</sup>

<sup>1</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

<sup>2</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Electrónica

[dm.gar93@gmail.com](mailto:dm.gar93@gmail.com), [auroravargast@hotmail.com](mailto:auroravargast@hotmail.com),  
[svergara2@hotmail.com](mailto:svergara2@hotmail.com)

**Resumen.** En el presente trabajo se presenta un avance del protocolo de investigación del tema de tesis que será propuesto ante el comité evaluador para su aprobación, se aborda la posibilidad de calcular parámetros propios de un robot cartesiano, tal como la masa, la fricción, etcétera, de manera automática utilizando técnicas de inteligencia artificial en un sistema mínimo.

**Palabras Clave:** Identificación, redes, neuronales.

### 1 Introducción

Los robots de configuración X,Y,Z pueden ser usados para distintos propósitos solo cambiando el actuador final, por ejemplo, impresora 3D, cortador laser, desbastadora, dispensadores de componentes, transportador de componentes peligrosos o delicados entre otros propósitos (Fernando Reyes Cortes 2011).

Hay varias etapas cruciales para la construcción de estos robots, una de ellas es la parte mecánica, se intenta tener la menor cantidad de tolerancias en cuanto a los engranes, tornillos, y sobre todo los rieles donde se desliza cada uno de los ejes tiene que estar perfectamente perpendicular a los otros dos ejes. Ya que este tipo de robots se tiene que mover de forma autónoma, es decir que se le dé una lista de puntos a recorrer y el robot los recorra con precisión y exactitud. Esto implica un control de lazo cerrado de cada uno de los motores que controla los ejes, y en algunos casos el control de temperatura del extrusor del material para imprimir en 3D, en otro caso la velocidad de la herramienta que

desbasta. Existen diferentes técnicas de control que se pueden usar para este tipo de aplicaciones (PD, PI, PID, Tangente hiperbólica, etc.), algo que tienen en común estas técnicas es que todas deben de ser sintonizadas, es decir encontrar las ganancias y los parámetros que se necesitan encontrar para que el control funcione como se espera, un ejemplo de estos parámetros es la ganancia máxima del controlador, es decir la potencia máxima que puede indicarle el control a un motor para que se mueva, la velocidad necesaria para que el material depositado pueda secarse en el caso de la impresora 3D. El cálculo de estos parámetros puede llevar desde horas hasta meses, esto se debe a que no se cuenta con una metodología con la cual se obtengan los parámetros en un tiempo definido, en muchos casos estos parámetros se encuentran a prueba y error.

## **2 Objetivo general y objetivos específicos de la investigación**

Objetivo general:

- El robot tipo X,Y,Z sea capaz de encontrar de manera automática sus parámetros.

Objetivos específicos:

- Crear una metodología capaz de calcular los parámetros del robot.
- El robot sea capaz de poner a prueba los parámetros encontrados y definir si son correctos.
- Crear una interfaz amigable para el uso del robot.

## **3 Pregunta(s) de investigación**

1. ¿Qué condiciones son necesarias para poder usar redes neuronales?
2. ¿Cómo se puede implementar el uso de redes neuronales entrenadas para encontrar los parámetros?
3. ¿Cuáles son las heurísticas y metaheurísticas que se pueden utilizar para la búsqueda de estos parámetros?

## **4 Hipótesis**

Es necesario entrenar la red neuronal que se pretende usar. Es necesario encontrar una manera de retroalimentar la red neuronal, esto permitirá que se ajusten los parámetros de cada nodo y de esta manera se pueda utilizar en diferentes robots del mismo tipo.

Alguna heurística puede aportar los pasos a seguir para encontrar dichos parámetros.

## 5 Límites de la tesis

El trabajo de tesis se dará por terminado cuando la metodología desarrollada logre encontrar automáticamente los parámetros de configuración del robot X,Y,Z y los parámetros encontrados sean puestos a prueba y el robot trabaje de manera precisa y exacta.

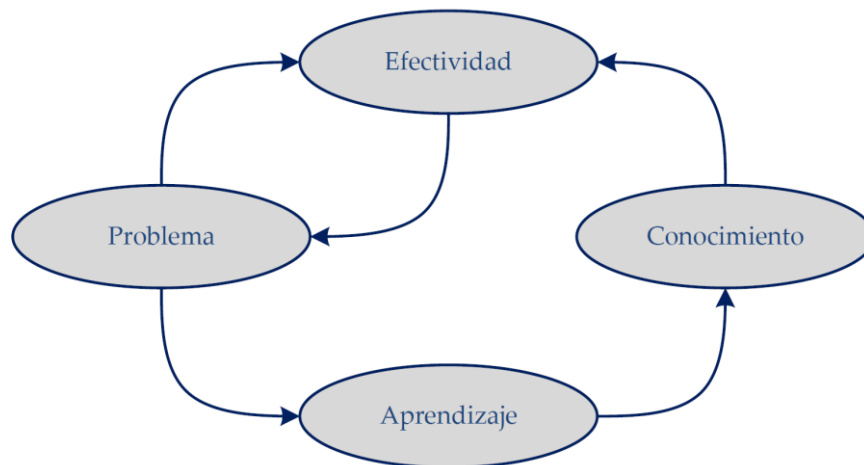
## 6 Marco teórico

La definición de inteligencia artificial varia de autor en autor, dado que es prácticamente imposible alcanzar una definición de I.A. con la que todo el mundo esté de acuerdo, Keith Downing propone que nos conformemos con una definición mucho más ambigua y menos ambiciosa: hacer lo correcto en el momento adecuado, desde el punto de vista de un observador humano externo (Keith L. Downing 2015). Otro punto de vista es el de Elaine Rich, de la Universidad de Texas en Austin: el estudio de cómo hacer que los ordenadores hagan cosas que, por ahora, los humanos hacemos mejor (Elaine Rich 1983). Ésta es una definición móvil de la I.A., ya que, en cuanto un problema abordado por técnicas de I.A. se resuelve satisfactoriamente, pasa a quedar automáticamente fuera de su ámbito de actuación. Por otro lado, mantiene perenne el interés de la I.A. centrando nuestra atención en aquellos problemas que, actualmente, los seres humanos somos capaces de resolver mejor que las máquinas.

### **Aprendizaje automático**

Tradicionalmente, si alguien quería resolver un problema con la ayuda de un ordenador, tenía que diseñar e implementar un algoritmo que especificase, hasta el más mínimo detalle, qué es lo que el ordenador tenía que hacer. Para algunos problemas, con multitud de casos particulares imposibles de prever, esta estrategia, simple y llanamente, no funciona.

El aprendizaje automático, o machine learning, proporciona mecanismos mediante los cuales el ordenador es capaz de aprender por sí mismo a resolver un problema. En estos casos, el programador se encarga de diseñar un algoritmo de aprendizaje que resulte adecuado para el problema que se pretende resolver, pero es el ordenador el que resuelve el problema, aprovechando para ello los datos a los que tenga acceso y las heurísticas de aprendizaje incorporadas en el algoritmo de aprendizaje creado por el programador. En cierto modo, el ordenador es capaz de programarse a sí mismo.



*Figura 1. Interacciones entre el problema, problema, el proceso de aprendizaje, el conocimiento adquirido y la efectividad observada en la resolución del problema.*

En Inteligencia Artificial, el aprendizaje se entiende como un proceso por el cual un ordenador es capaz de mejorar su habilidad en la resolución de un problema a través de la adquisición de conocimiento, conocimiento que obtiene a través de la experiencia (Thomas Mitchell 1997).

¿Cómo puede adquirir experiencia un sistema automatizado? A partir de los datos que recibe. Un neurocientífico catalogaría como aprendizaje cualquier mecanismo que utilice las entradas presentes para dar forma a futuras respuestas; esto es, cualquier mecanismo que proyecte los datos de los que dispone en el presente hacia el futuro para cambiar su comportamiento. Ese cambio de comportamiento se realiza gracias a un proceso de entrenamiento con los datos disponibles, que le permiten a un sistema anticipar lo que puede suceder en el futuro. El entrenamiento puede realizarse de antemano (entrenamiento offline), de tal forma que el sistema se entrena antes de ponerlo en marcha en el entorno en el que desempeñará su labor. O también puede ir realizándose sobre la marcha (entrenamiento online), conforme el sistema realiza la tarea para la que ha sido diseñado. En el primer caso, su comportamiento se mantendrá estable a lo largo del tiempo, lo que nos permite predecir su rendimiento en un entorno controlado. En el segundo caso, su comportamiento se adaptará a los cambios que se produzcan en su entorno, lo que puede resultar impredecible.

#### **Sintonización de controladores**

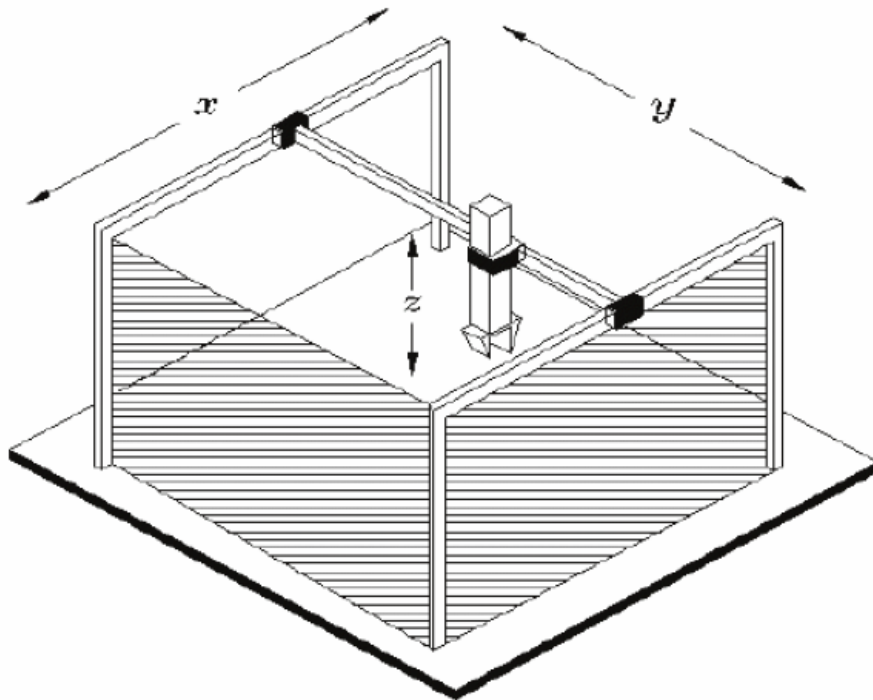
Sintonizar un sistema de control realimentado significa regular parámetros en el controlador para lograr implementar un control robusto en el proceso. “Robusto” en este contexto es usualmente definido como la estabilidad de las variables de procesos a pesar de los cambios de carga, una rápida respuesta ante los cambios de setpoint, oscilaciones mínimas y un offset mínimo (error entre el setpoint y la variable de proceso) en el tiempo.

Diferentes tipos de procesos, que tienen diferentes comportamientos dinámicos (dependientes del tiempo), requieren diferentes niveles de acción de control proporcional, integral y derivativo para lograr una respuesta estable y robusta. Y por tanto alguien que intente o busque sintonizar un controlador PID debe entender la naturaleza dinámica del proceso que está siendo controlado.

### **Robot cartesiano**

Los robots manipuladores cuyas primeras tres articulaciones son prismáticas se les denomina robots cartesianos (Fernando Reyes Cortes 2011), y este tipo de robots también son conocidos como robots lineales o “X, Y, Z”.

Los ejes asociados a cada articulación X, Y y Z son mutuamente perpendiculares entre sí. En la figura 2 se muestra un prototipo de un robot cartesiano. El extremo final del robot se programa en coordenadas cartesianas, el espacio de trabajo de este manipulador es un paralelepípedo recto y su estructura mecánica presenta baja destreza debido a que todas sus articulaciones son prismáticas.



*Figura 2. Espacio de trabajo del robot cartesiano.*

## 7 Diseño metodológico

La investigación por realizar es del tipo cuantitativa debido a la naturaleza del problema, el resultado esperado es la obtención de los valores únicos de los parámetros desconocidos de un robot. Dado que se busca un resultado único se contempla una investigación descriptiva, esto nos permite llegar a un único resultado, y es posible medir la exactitud de este, se definen los objetivos a alcanzar y la metodología para llegar a ellos, aunque no se sabe si se lograra el resultado deseado.

Tanto la investigación, como la experimentación se llevará a cabo en el laboratorio de robótica de la maestría en ciencias de la electrónica, debido a que en este laboratorio se encuentra el robot disponible con el que se va a trabajar.

El robot con el que se va a trabajar es un robot de tres ejes (X,Y y Z) que se muestra en la figura 3, esta selección es por conveniencia ya que se cuenta con las facilidades para el uso y manipulación del robot. El robot cuenta con la mecánica necesaria para el movimiento en tres dimensiones, el movimiento de cada eje es producido por un motor de corriente directa que es controlado individualmente, cada motor cuenta con un encoder acoplado para poder conocer la posición del actuador final en todo momento, para el control del robot se utiliza un sistema embebido mostrado en la figura 4, con un FPGA, memoria RAM, memoria ROM, el cual cuenta con un microprocesador, el cual se programa con un lenguaje de alto nivel propio, esto permite la implementación de los algoritmos desarrollador en tiempo real en el robot. Para desarrollar una metodología para encontrar los parámetros intrínsecos del robot no se requiere de más población de estudio ya que la metodología debe de funcionar en cualquier robot con la misma estructura del robot de estudio.

Los datos experimentales se obtendrán directamente del robot, variables como: posición, velocidad, aceleración, con el fin de comparar los valores de los parámetros obtenidos si es necesario se medirán físicamente variables como: volumen, masa, distancias, etcétera.

Los resultados que se obtengan servirán para ajustar la metodología propuesta y así obtener un mejor porcentaje de exactitud en la búsqueda de los parámetros del robot.

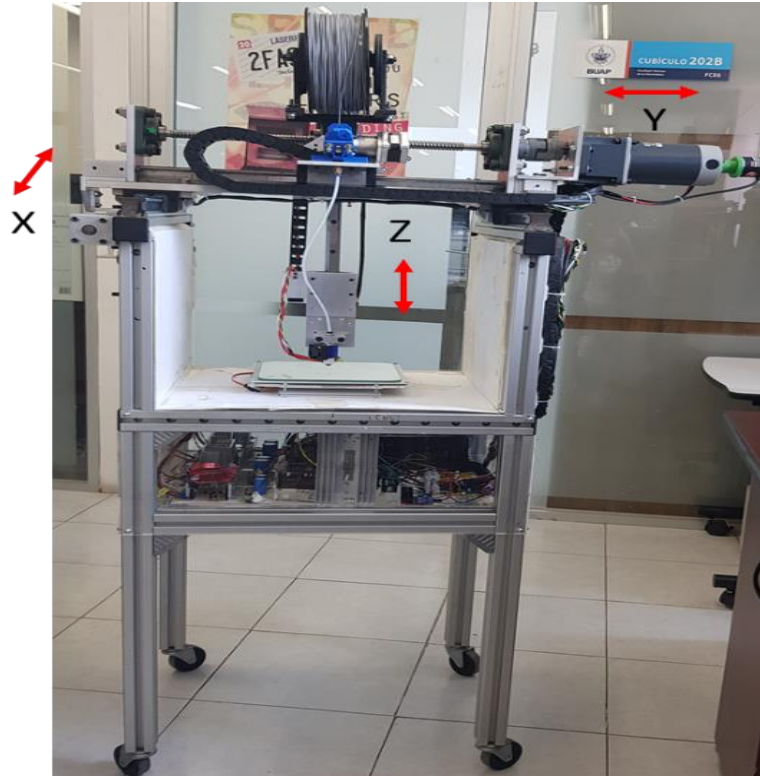


Figura 3. Robot de 3 grados de libertad.

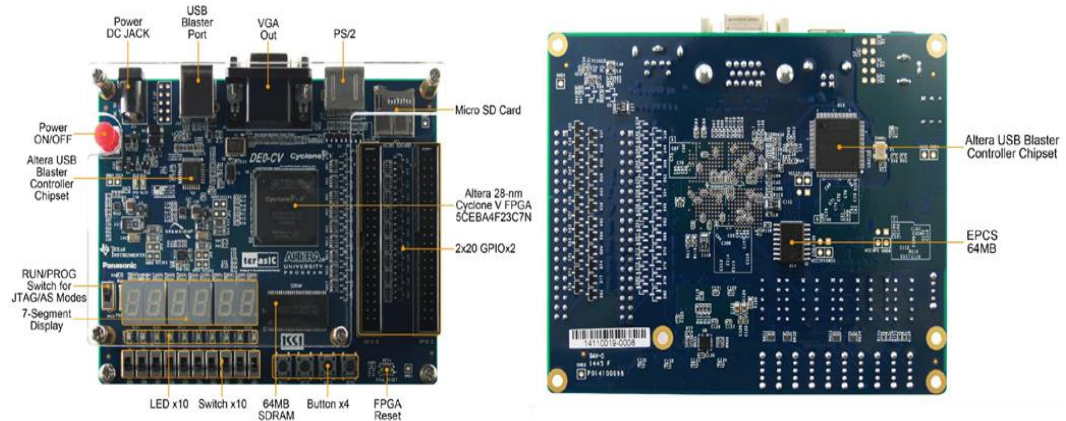


Figura 4. Tarjeta DE0-CV marca Terasic.



## Referencias

- Elaine Rich (1983). "What is Artificial Intelligence?" en Elaine Rich (1983). "Artificial Intelligence." (pp. 1-24) McGraw-Hill.
- Fernando Reyes Cortes (2011). "Cinémática de robots manipuladores" en Fernando Reyes Cortes (2011). "Robótica control de robots manipuladores" (pp 243-245) ALFAOMEGA
- Keith L. Downing (2015). "Recognizing Emergent Intelligence." en Keith L. Downing (2015). "Intelligence Emerging: Adaptivity and Search in Evolving Neural Systems." (pp. 365-408) MIT Press.
- Thomas Mitchell (1997). "Concept Learning and the General-to-Specific Ordering" en Thomas Mitchell (1997). "Machine Learning" (pp 20-50) McGraw-Hill

# Capítulo 14

## Caso: Identificación de Transacciones de Criptomonedas de Delitos Cibernéticos en Tor

Julio Jesús Salas Conde, Manuel Martín Ortíz

Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Computación

`jsalasconde@gmail.com`, `mmartinmx@gmail.com`

**Resumen.** Este artículo pretende ser un caso de estudio para que a partir de las cuentas o billeteras de Blockchain que se publican en redes onion con contenido delictivo de alto impacto como lo son la venta de armas, de drogas y estupefacientes, la trata de personas y delitos contra menores se pueda saber si han tenido transacciones y cuentan con montos de entrada y de salida, utilizando técnicas y herramientas de identificación, recuperación de información, y análisis para ello, esta información puede ser de valor para las agencias de seguridad en México en su persecución de los delitos de alto impacto. Además se identifican algunas técnicas que usan los administradores de estas redes onion para evitar tener una sola cuenta concertadora para la red onion y así complicar el seguimiento de las transacciones por Blockchain.

**Palabras Clave:** Wallet, Blockchain, delitos, Tor, drogas, trata, menores, armas, onion, Dark Web.

### 1. Introducción

En diciembre de 2019, la Agencia del Departamento de justicia de los Estados Unidos dedicada a la lucha contra el contrabando y el consumo de drogas, además del lavado de activos DEA (Drug Enforcement Administration), publicó un documento acerca de su evaluación nacional sobre las amenazas de las drogas en Estados Unidos, en la cuál se trata el tema de financiamiento ilícito a través de monedas virtuales como es el Bitcoin entre otras más (DEA, 2019), esto generó expectativa en los medios de comunicación de México debido a que organizaciones criminales transnacionales mexicanas sirven como proveedores de drogas a dicho país, señalando que organizaciones de tráfico independientes más pequeñas compran Fentanilo a China en la Web abierta y la venden a EU en la web oscura.

En la actualidad es de vital importancia el estudio de las redes privadas, ya que es conocido el uso de las mismas para la comisión de delitos en los ámbitos informáticos, por

ejemplo se puede citar que existen delitos cibernéticos en redes Tor, los cuáles requieren de un análisis para ser identificados y proporcionar información relevante a los investigadores.

Además es importante destacar que no solamente esto sucede en la red Tor, existen otras redes como Ares, Freenet, I2P, en la Deep Web, sin embargo esta es la más usada.

Existen técnicas intrusivas para identificar y deshabilitar sitios en la Dark Web como botnets, malware y penetración a sistemas sin consentimiento, sin embargo estas son consideradas conductas delictivas en México, por lo que la investigación debe realizarse a partir de la inteligencia generada con los datos publicados en estas mismas correlacionándolas con datos públicos, haciendo uso de técnicas de reconocimiento de patrones, minería de datos, aprendizaje de máquina, que nos puedan asociar y descubrir información como el reconocimiento de rostros, la identificación de datos duros como nombres, direcciones, alias, correos electrónicos, números telefónicos, entre otras mas que se permitan.

Por lo tanto, es necesario que las autoridades correspondientes investiguen los delitos cibernéticos en la Dark Web utilizando estas técnicas para mejorar el proceso de investigación en un caso, así como también prevenir estos delitos al monitorear estas conductas en estas redes privadas.

## **2. Estado del Arte**

(Hurlburt, 2017) La raíz de muchas, si no la mayoría, de las amenazas a la ciberseguridad no está en el borde de Internet, sino en su interior, en la Dark Web. Sin embargo, la Dark Web es cada vez más difícil de descifrar a medida que las técnicas de privacidad y encriptación se vuelven más sofisticadas. Los sitios serán mucho menos visibles y solo se podrá acceder mediante invitación. Además, Bitcoin, una vez que la criptomoneda elegida en Dark Web, está siendo rápidamente reemplazada por Monero, que ofrece mecanismos sigilosos que evitan el rastreo indirecto de aquellos que realizan transacciones, una vulnerabilidad que ha afectado al bitcoin. Las mismas herramientas de código abierto promocionadas por los defensores de la privacidad para proteger los datos personales y eludir la censura y la vigilancia del gobierno también están alimentando la actividad delictiva generalizada.

Dada la sofisticada infraestructura de Dark Web y las capacidades técnicas superiores de muchos de sus habitantes, es poco probable que las técnicas forenses tradicionales tengan un efecto sustancial o duradero. Sin embargo, las nuevas herramientas de aprendizaje automático, minería de datos y análisis están a punto de convertirse en formidables armas ofensivas en la lucha contra el delito cibernético.

Según (BrightPlanet, 2013), el Internet se construye alrededor de páginas web que hacen referencia a otras páginas, si se tiene una página web de destino que no tiene enlaces entrantes, se ha ocultado esa página y no puede ser encontrada por usuarios o motores de búsqueda (no está publicada con un link). Un ejemplo de esto sería una publicación de un blog que aún no se ha indexado. La publicación del blog puede existir en el Internet público, pero a menos que se conozca la URL exacta, nunca se encontrará.

La Surface Web es una parte de Internet que se puede hallar a través de técnicas de rastreo de enlaces, conocida como Link-crawling, que significa que los datos enlazados se pueden encontrar a través de un hipervínculo desde la página principal de un dominio y el buscador puede extraer estos datos.

La Deep Web es una parte de Internet no accesible a los motores de búsqueda de rastreo de enlaces como Google, Yahoo y Bing.

La Dark Web es una parte de la World Wide Web que necesita un tipo especial de software para acceder y se refiere específicamente a una colección de sitios Web que existe en una red cifrada a la que no se puede acceder mediante los motores de búsqueda tradicionales o incluso que visitan los navegadores Web tradicionales. Una vez que esté dentro de la Dark Web, se puede acceder a los sitios Web y otros servicios a través de un navegador de la misma manera que en una Web tradicional. Sin embargo, hay algunos sitios que están ocultos de manera efectiva, lo que significa que tradicionalmente no han sido indexados por un motor de búsqueda y, por lo tanto, solo se puede acceder a dichos sitios si conoce la dirección del sitio (Rafiuddin, 2017).

Las redes privadas virtuales son otro aspecto de la Deep Web, que existe dentro del Internet público, y a menudo requieren software adicional para accederlas. TOR (The Onion Router) es un gran ejemplo. Oculto dentro de la red pública está esta red privada de contenido diferente y a la que solo se puede acceder mediante la red TOR.

### **3. Caso de Transacciones en Criptomonedas en delitos de alto impacto**

(DEA, 2019) Los métodos preferidos para mover y lavar ingresos ilícitos se han mantenido en gran medida iguales a lo largo de los años. Durante varios años la moneda virtual se ha utilizado como un método de pago común para comprar drogas ilegales en línea, ahora los lavadores de dinero internacionales lo utilizan cada vez más para transferir ingresos a través de las fronteras en nombre de las TCO (Transnational Criminal Organizations / Organizaciones criminales transnacionales).

Los métodos comunes de lavado de dinero utilizados para trasladar los ingresos por drogas a fuentes de suministro extranjeras, generalmente involucran una red de corredores de dinero involucrados en un híbrido de actividades TBML (Trade-Based Money Laundering / Lavado de dinero basado en el comercio), incluido BMPE (Black Market Peso Exchange / Mercado negro de cambio de pesos).

Los MLO (Money Laundering Organizations / Organizaciones de Lavado de Dinero) asiáticos buscan beneficiarse de las actividades ilícitas asociadas con las TCO mexicanas y colombianas, así como de la reventa de dólares estadounidenses en los Estados Unidos a ciudadanos chinos que buscan evadir las estrictas leyes de control de divisas de China.

Estas restricciones han llevado a la evolución de un mercado negro informal que permite a los ciudadanos chinos sacar dinero de China mediante el intercambio de activos con base en China por divisas u otros activos, como los ingresos de las drogas, ubicados en el extranjero. La participación de los MLO asiáticos en el movimiento de las ganancias de drogas ilícitas también ha aumentado en los últimos años, ya que las DTO han tratado de simplificar la adquisición y el pago de los envíos de precursores químicos.

Descripción	Imagen del sitio
<p><b>Agartha Underground Oasis.</b></p> <p>Agartha es un mercado anónimo. Esto significa que sus usuarios esperan y tienen derecho al anonimato. Los proveedores deben destruir toda la información del cliente al completar cada pedido.</p> <p>Costo del bono del proveedor: 0.04 BTC.</p> <p>Productos prohibidos: Sin fentanilo, heroína, metanfetamina, crack, MDPV, DNP, escopolamina o THC / especias sintéticas.</p> 	<p><b>Empire Market.</b></p> <p>Una vez que compre, se marcará como procesamiento, el proveedor tiene 48 horas para completar el pedido enviando el artículo digital y luego marcando el pedido como enviado, de lo contrario, el pedido se cancelará. Una vez que el pedido se ha marcado como enviado, el comprador puede finalizar el pedido una vez que ha recibido la compra, disputar el pedido si hay un problema o extender el depósito en garantía si es necesario. Si el comprador no hace nada una vez que el pedido está marcado como enviado, finalizará automáticamente después de haber sido marcado como enviado durante 48 horas.</p> 

Tabla 1. Ejemplos de sitios de venta de drogas

Los métodos de lavado de dinero utilizados por los DTO nacionales generalmente involucran negocios intensivos en efectivo, así como los sectores de importación-exportación, servicios de dinero MSB (Money Service Business / Negocios de Servicio de Dinero), casinos y bienes raíces. Los métodos típicos de lavado de dinero incluyen:

- Compra, depósito y uso de efectivo e instrumentos monetarios en montos bajo los límites regulatorios;
- Abrir cuentas bancarias y de corretaje utilizando corporaciones fantasma y nomina-dos para disfrazar la identidad de las personas que controlan las cuentas;
- El uso de empresas fachada para combinar efectivo de drogas con ingresos lícitos para dar a las ganancias de la droga la apariencia de una fuente legítima;
- Explotación de productos y servicios bancarios, a veces como resultado del cumpli-miento deficiente de las obligaciones AML (Anti-Money Laundering / Anti lavado de dinero).
- Facilitación de actividades ilegales por parte de comerciantes e instituciones financie-ras, ya sean involuntarias o cómplices;

- Movimiento y colocación de fondos a través de bancos, MSBs con licencia, transmisores de dinero sin licencia y contrabando de efectivo;
- La evolución de las amenazas del cibercrimen, el robo de identidad y los nuevos sistemas de pago, como las monedas virtuales.

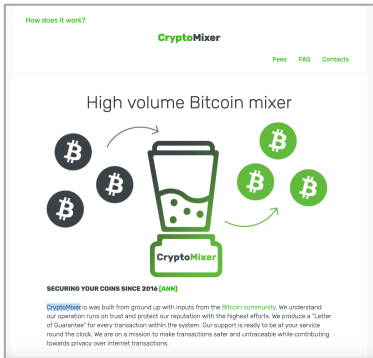

Descripción	Imagen del sitio
<p><b>CryptoMixer.</b></p> <p>Mantenga sus fondos imposibles de rastrear con el confiable mezclador de Bitcoin.</p> <p>Tenemos la misión de hacer que las transacciones sean más seguras e imposibles de rastrear, al tiempo que contribuimos a la privacidad en las transacciones por Internet.</p> <p>Funciona combinando sus monedas con las de nuestras reservas de criptomonedas para garantizar transacciones privadas sin dejar rastro.</p> <p>CryptoMixer tiene más de 2000 BTC en sus reservas de criptomonedas, tarifa desde 0.0005 BTC por transacción.</p> 	<p><b>Bitcoin Fog.</b></p> <p>El procedimiento estándar para anonimizar Bitcoins es hacer un depósito en la página de depósito y, después de un tiempo, solicitar un retiro.</p> <p>Sus Bitcoins se mezclan en nuestro grupo interno con los Bitcoins de otros usuarios y luego se les devuelve a su nueva dirección o varias direcciones en una cantidad de cantidades aleatorias en momentos aleatorios, lo que hace que sea muy poco probable rastrear el origen de sus Bitcoins y definitivamente es imposible de probar.</p> <p>El servicio actualmente solo cobra entre 1% y 3% de tarifa en todos los depósitos. La tarifa se asigna al azar para cada depósito para ocultar aún más el dinero de enlace que entra y sale de la niebla.</p> 

Tabla 2. Servicios de mezcla de Criptomonedas.

Tomando este último punto, a medida que ha aumentado el uso de monedas virtuales, han surgido servicios de intercambio para ayudar a convertir la moneda fiduciaria en Bitcoin y viceversa.

Los mercados anónimos de la web oscura, las transacciones en monedas virtuales siguen siendo una fuente de drogas ilícitas y otros productos de contrabando (ver Tabla 1).

El lavado de dinero en los Estados Unidos relacionado con estos mercados web oscuros, ocurre como resultado de que los proveedores acumulen Bitcoins u otras monedas virtuales y necesitan convertirlos en moneda fiduciaria.

Descripción	Imagen del sitio
<p><b>AgoraDesk.</b></p> <p>Mercado donde los usuarios pueden comprar y vender criptomonedas entre ellos.</p> <p>Los usuarios, llamados comerciantes, crean anuncios con el precio y el método de pago que desean ofrecer.</p> <p>Encontrará comerciantes que compran y venden criptomonedas en línea por más de 60 métodos de pago diferentes.</p> 	<p><b>Bitcoin Wallet.</b></p> <p>Simplemente cree una cuenta gratuita. Después de eso, obtendrá una nueva dirección de Bitcoin. Para pagar de forma segura, debe enviar Bitcoins a esa dirección, después de tener un saldo positivo, puede usar estos Bitcoins para pagar de forma segura otros servicios de Darknet (CP, Drogas, Armas).</p> <p>Sus transacciones en nuestro sitio no son rastreables, mientras que las billeteras Clearnet son rastreables. Mezclaremos los Bitcoins por usted, sin cargo.</p> <p>Tarifas de transacción en la red Bitcoin que actualmente está configurada en 0.001 ₿.</p> 

Tabla 3. Intercambiadores de Criptomonedas en Web oscura.

Los vendedores o compradores menos sofisticados enviarán criptomonedas directamente hacia y/o desde un mercado a un intermediador, dando como resultado una transacción directa. (ejemplos de estos intermediadores son Bitso, Plus500, Pay4y, Changelly, Binance, Etoro, Coinmama, Kraken, BcBitcoin, entre otros). Cabe señalar que estos sitios son rastreables por las autoridades debido a que se encuentran en la red indexada de Internet y cuentan con IPs públicas. En el sitio <https://coinmarketcap.com/> podemos encontrar las 100 principales criptomonedas por capitalización bursátil y los 100 principales intercambios de Criptomonedas por volumen comercial.

Otros vendedores o compradores usan servicios de mezcla de Bitcoins que ofuscan los orígenes de los fondos antes de enviarlos a los intermediadores. En la tabla 2, podemos ver ejemplos de estos servicios.

Debido a que la mayoría de los intermediadores de EE. UU. cumplen con las regulaciones, los delincuentes buscan intermediadores sin licencia e intermediadores de igual a igual, o se convierten en uno. En la tabla 3, podemos ver ejemplos de estos.

También pueden usar monedas virtuales menos transparentes o Criptomonedas de Ano-

nimato Mejoradas (AEC), para inhibir la capacidad de los investigadores de seguir los fondos a los intercambios identificados.

#### 4. Procedimiento de identificación de transacciones en cuentas con delitos de alto impacto

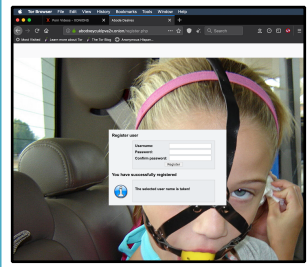
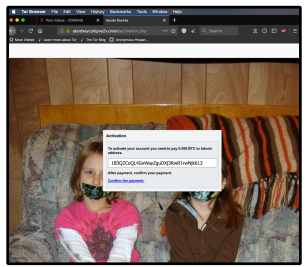
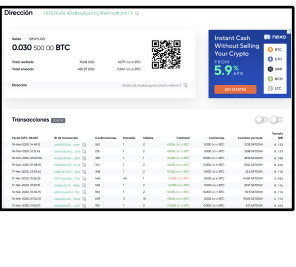
En base al punto de tener la capacidad de seguir los fondos a los intercambios identificados a través del Blockchain, se realizó un procedimiento que a partir de identificar las páginas con conductas delictivas de alto impacto, se obtuvieran sus cuentas (Wallets) a las cuales se verifica si tienen movimientos de entrada y salida, junto con sus transacciones, y sus montos.

Esto nos da valiosa información, como es saber si estas conductas delictivas han sido usadas a través de estos sitios o no, deducir el número de veces que han sido usadas, la fecha de inicio y de termino de esta conducta, los montos que se han recibido y pagado, entre otros puntos mas.

Como ejemplo, podemos apreciar en la figura 1, una red onion donde se muestra la conducta delictiva de tortura a menores de edad, al activar la cuenta pide realizar un pago de 0.006 Bitcoins a la chequera “1B3Q2CoQL4GxWayZguDXj3RwR1rwNjk613” y confirmar su pago como vemos en la figura 2.

Con esta cuenta podemos apreciar en la figura 3, que tiene un total de 0.077 Bitcoins recibidos equivalente a 741.18 dólares en un periodo del 02 al 18 de febrero del 2020.

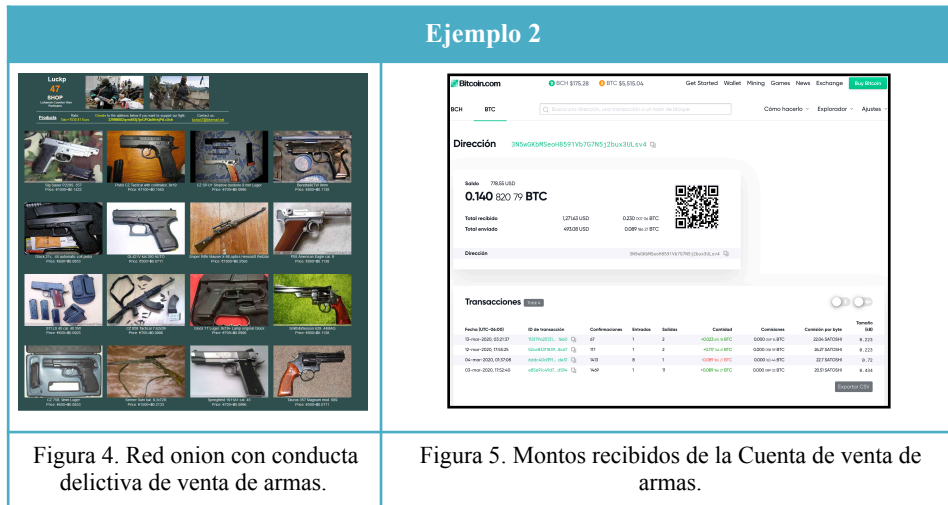
**Ejemplo 1**

		
<p>Figura 1. Red onion con conducta delictiva de tortura.</p>	<p>Figura 2. Cuenta para realizar pago.</p>	<p>Figura 3. Transacciones y montos realizados por la Cuenta.</p>

En el caso de la conducta delictiva de venta de armas, podemos ver en la figura 4, una red onion que ofrece armas cortas, largas, y municiones a nivel mundial, el pago se realiza en Bitcoins a la billetera “3N5wGKbMSeoH8591Vb7G7N5j2bux3ULsv4”, al analizar los



movimientos de esta cuenta en el Blockchain, podemos ver en la figura 5 que presenta un total recibido de 1,271.63 dólares en el período del 03 al 13 de marzo de 2020, por tres transacciones, también se puede notar que existe una transacción de salida por 493.08 dólares.



En el caso de venta de drogas y estupefacientes, podemos ver en el ejemplo 3, un sitio web de la red onion de venta de drogas, como vemos en la figura 6, Al realizar la orden de un producto podemos ver en la figura 7 la billetera a la cual se pide que se realice el pago “1LF4oZqQcP6p3gx4tVd26GfaqY8E21H4ve”, posteriormente en la figura 8, vemos que no existen montos ni transacciones realizados en esta cuenta.

Como se puede ver en la imagen 8, no todas las cuentas presentan saldo o transacciones, en este caso al volver a realizar el pedido por el mismo producto cambia la cuenta o billetera, por lo que esto supone un problema al seguimiento en Blockchain, ya que el administrador del sitio genera una billetera por cada venta que realiza.

De la misma forma en otras redes onion de venta de droga usan otras técnicas para evitar el seguimiento en el Blockchain de una sola billetera concentradora, ver figura 9, donde al solicitar un producto de droga pide que se depositen los fondos a una billetera de nueva creación la cuál al registro del usuario se le asigna para sus compras.

### Ejemplo 3

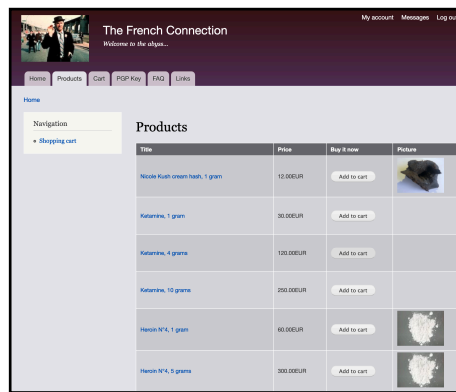


Figura 6. Red onion con conducta delictiva de venta de drogas.

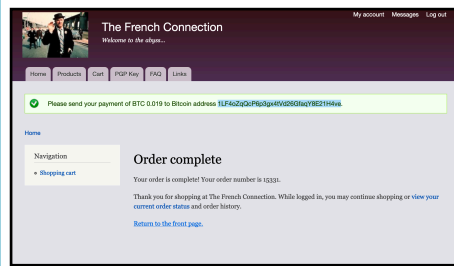


Figura 7. Cuenta para realizar pago.

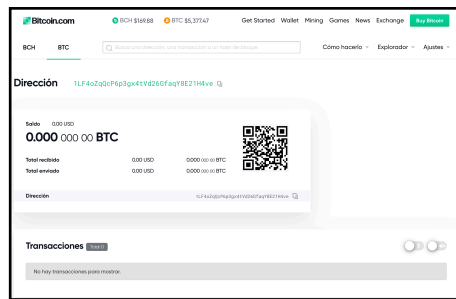


Figura 8. Transacciones y montos realizados por la Cuenta.

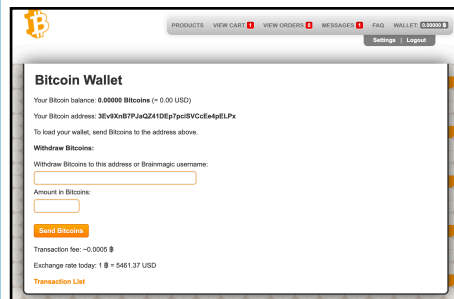


Figura 9. Billetera asignada al usuario registrado para compra.

## 5. Conclusiones

Se han utilizado técnicas y herramientas de identificación, recuperación de información y análisis para analizar alrededor de 2,000 redes onion, de las cuales un promedio del 51% contienen alguna conducta delictiva, de entre ellas se identifican delitos contra la salud, contra la libertad de tránsito, contra la libertad sexual, contra la seguridad nacional, contra la seguridad financiera y contra las tecnologías de la información. Se dio seguimiento a 60 redes de tráfico de narcóticos, 75 redes de delitos contra menores y trata, y 59 redes onion de venta de armas, esto con el fin de identificar sus billeteras y comprobar si

existen transacciones y montos en ellas. No se realizó ninguna compra o adquisición de algún producto o servicio, ni el almacenaje de conductas sexuales en menores y mayores de edad, y se tomaron las medidas de seguridad pertinentes como lo es uso de VPN segura, desactivación del uso de Scripts en las redes, no proporcionar datos confidenciales ni correos electrónicos de la red indexada, conexión desde usb live, no tener ninguna comunicación directa, entre otras. Se puede concluir que aunque las redes privadas virtuales como TOR son anónimas ya que ocultan su IP real, existen otros métodos que nos pueden arrojar información relevante para los investigadores de delitos de alto impacto y las agencias de seguridad.

## Referencias

- Alnabulsi, H., & Islam, R. (2018, December). Identification of Illegal Forum Activities Inside the Dark Net. IEEE.
- BrightPlanet. (March de 2013). UNDERSTANDING THE DEEP WEB IN 10 MINUTES.
- Cheong, F. C. (1996). Internet Agents: Spiders, Wanderers, Brokers, and Bots. Indianapolis.
- DEA, (2019). National Drug Threat Assessment.
- Hurlburt, G. (26 de April de 2017). Shining Light on the Dark Web. IEEE, 50.
- Pant, G. S. (2002). Exploration versus Exploitation in Topic Driven Crawlers. In Proceedings of the WWW Workshop on Web Dynamics.
- Rafiuiddin, M. F. B., Minhas, H., & Dhubb, P. S. (2017, September). A dark web story in-depth research and study conducted on the dark web based on forensic computing and security in Malaysia. IEEE.

# Capítulo 15

## Análisis sintáctico de $HRG's$

Yolanda Moyao<sup>1</sup>, Darnes Vilariño<sup>1</sup>, Carlos Guillén<sup>2</sup> and José de Jesús Lavalle<sup>1</sup>

<sup>1</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

<sup>2</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias Físico Matemáticas

{ymoyao,dvilarino,jlavalle}@cs.buap.mx, cguillen@cfm.buap.mx

**Resumen.** En este trabajo se presenta un algoritmo, para resolver el problema del análisis sintáctico de la Gramática de Reemplazo de Hiperaristas  $G$ , dado un *hipergrafo* enraizado  $H$ . El problema consiste en determinar si  $H \in L(G)$ , es decir ¿ $H$  está en el lenguaje generado por  $G$ ? A diferencia del algoritmo propuesto por (Chiang et al., 2013), nuestro algoritmo evita el cálculo de la buena descomposición hiperarbórea, mejorando de esta manera su implementación.

**Palabras clave:** Gramáticas de reemplazo de hiperaristas, Hipergrafo, Analisis sintáctico.

### 1 Introducción

Las gramáticas grafo han sido desarrolladas como una extensión del concepto formal de gramáticas en cadenas a gramáticas en hipergrafos. Una forma para describir los lenguajes de hipergrafos es a través de las Gramáticas de Reemplazo de Hiperaristas ( $HRG's$ ) por sus siglas en inglés, las cuales son un tipo de gramáticas grafo, éstas han sido utilizadas en algunas aplicaciones tales como, semántica de sistemas de bases de datos de lenguajes de programación, reconocimiento de patrones Habel (1991).

El problema de decidir si un hipergrafo es generado por una  $HRG$  en general es  $NP$ , Lautemann (1990), Drewes et al. (1997) debido a que hay una correspondencia lineal entre el tamaño  $n$  de  $H$  y el número de pasos de derivación requeridos para derivar  $H$  de  $G$ , es decir que para cada gramática hay una función lineal  $f$  tal que, para cada hipergrafo  $H$  de tamaño  $n$ , Si  $H \in L(HRG)$  entonces  $H$  tiene una derivación de longitud a lo más  $f(n)$ . Sin embargo se han presentado algunos resultados que resuelven este problema en tiempo polinomial.

(Lautemann, 1990) presenta una versión extendida "Top-Down" para  $HRG's$  del algoritmo CKY de grámaticas libres de contexto para cadenas. También analiza algunas condiciones para las  $HRG's$  o para los grafos de entrada, las cuales garantizan un analizador en tiempo polinomial. Sin embargo el analizador lo presenta bajo un nivel muy abstracto que no permite la implementación de manera directa y tampoco presenta un análisis detallado de la complejidad del algoritmo.

Por otro lado Chiang et al. (2013) presenta un analizador "Bottom-Up" para  $HRG's$  representado a través de un sistema deductivo, también presenta un esquema de optimización, el cual permite ejecutar el algoritmo analizador en tiempo polinomial, cuando el ancho y el grado del árbol del hipergrafo sean acotados.

En este trabajo se presenta una solución para el análisis sintáctico de *HRGs*, dicho análisis consiste de hacer coincidir las reglas de producción correspondientes que cubren todas las partes del hipergrafo de entrada *H*, es decir identificar a cada uno de los subhipergrafos.

La contribución principal de este trabajo de investigación es un algoritmo simple, pues evita el cálculo de la buena descomposición arbórea y además eficiente para resolver el problema de análisis sintáctico de las *HRGs*.

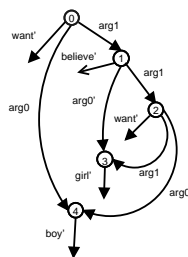
El artículo se encuentra organizado de la siguiente manera, en la sección 2 se abordan algunos conceptos tales como, *HRGs*, Hipergrafo, derivación, lenguaje de hipergrafo e hipergrafo isomorfo necesarios para el desarrollo de éste trabajo de investigación. En la sección 3 se describe el proceso de un analizador sintáctico junto con el desarrollo de un ejemplo que muestra dicho proceso y finalmente en la sección 4 se muestran las conclusiones de este trabajo de investigación.

## 2 Gramáticas de Reemplazo de Hiperarista (*HRGs*)

La Representación de Significado Abstracto (*AMR*) por sus siglas en inglés son un formalismo para la representación semántica de las sentencias que conforman un lenguaje natural, donde el significado de una oración es codificado a través de Grafos acíclicos, dirigidos (*DAGs*), por sus siglas en inglés, enraizados y etiquetados (Flanigan et al., 2014).

En este formalismo se tienen dos tipos de aristas, las que son hojas y las que no lo son. Las aristas hojas, representan conceptos o entidades, los cuales pueden significar palabras o palabras clave, las aristas que no son hojas representan la relación o rol entre conceptos conectados entre sí, la cual está determinada por los marcos semánticos o instancias (Banarescu et al., 2013), (Peng et al., 2015).

En la Figura 1 se muestra un ejemplo de un *AMR*, cuyo lenguaje sobre el que se basa está formado por dos marcos semánticos (want y believe), dos tipos de entidades (boy y girl) y dos roles (arg0 y arg1).



**Fig. 1.** AMR que representa el significado "The boy wants the girl to believe that he wants her" (Referenciada de Peng et al. (2015))

Para representar las *AMR* se usan hipergrafos, cuyas aristas reciben el nombre de hiperaristas y éstas pueden relacionar cualquier cantidad de vértices, a diferencia de sólo

un máximo de dos como sucede para los grafos, los cuales son un caso particular de los hipergrafos. Se dice que un hipergrafo es etiquetado, si cada una de sus hiperaristas tienen asociado algún dato. A través de las *HRGs* se pueden generar lenguajes de hipergrafos (Chiang et al., 2013), donde el lenguaje resultante consiste de grafos de cadenas, es decir, hipergrafos que modelan cadenas.

**Definición 1** *Un hipergrafo  $H$  sobre  $C$  es una tupla  $H = (V, E, lab, att, ext)$  (Drewes et al., 1997) donde*

- $V$  es un conjunto finito de nodos.
- $E$  Es un conjunto finito de hiperaristas.
- $att : E \rightarrow V^*$  es un mapeo que asigna por pares una secuencia de nodos de conexión distintos  $att(e)$  a cada  $e \in E$ .
- $lab : E \rightarrow C$  es un mapeo que etiqueta cada hiperarista tal que  $type(lab(e)) = |att(e)|$ , donde  $type(e) = |e|$ , es decir representa el número de vértices conectados a cada hiperarista.
- $ext \in V^*$  es una secuencia por pares de nodos distintos, llamado conjunto de nodos externos.

**Definición 2** *Sea  $G, H$  un hipergrafo. Decimos que  $H$  es un subhipergrafo de  $G$  si  $V_H \subseteq V_G, E_H \subseteq E_G, att_H \subseteq att_G$ , y  $\forall at \in att_H, lab_G(at)$  y  $ext_H(at) = ext_G(at)$  (Lautemann, 1990)*

La Figura 2 inciso a) se muestra un hipergrafo, donde  $V = \{1, 2, 3, 4\}, E = \{(1), (1, 2), (1, 3), (2, 3, 4)\}, C = \{believe', arg0, arg1, girl', Y\}, att(\{(1)\}) = 1, att(\{(1, 2)\}) = 2, att(\{(1, 3)\}) = 2, att(\{(2, 3, 4)\}) = 3$

Las *HRGs* son elementos formales que pueden ser empleados para la generación y análisis de la representación semántica apoyada en hipergrafos (Groschwitz et al., 2015), de hecho se consideran una herramienta efectiva para resolver problemas de análisis y generación automática de lenguajes libres de contexto, a través del reemplazo de hiperaristas, es decir, las *HRGs* se consideran como un mecanismo de reescritura de hipergrafos, éstas tienen la ventaja de que son una extensión natural de las gramáticas libres de contexto para cadenas y para grafos.

**Definición 3** *Una Gramática de Reemplazo de Hiperaristas (*HRGs*) es una tupla  $G = (N, T, S, P)$  (Aguinaga et al., 2018) donde,*

- $N$  Es un conjunto finito de símbolos no terminales. Cada no terminal  $A$  tiene un rango entero no negativo, el cual se escribe como  $|A|$ .
- $T$  es un conjunto finito de símbolos terminales.
- $S \in N$  es un símbolo no terminal inicial,  $|S| = 0$ .
- $P$  es un conjunto finito de reglas de producción de la forma  $R_i : A \rightarrow h$ , donde  $A \in N$ , y  $h$  es un hipergrafo cuyas aristas están etiquetadas por símbolos de  $T \cup N$ . Si una arista  $e$  es etiquetada por un no terminal  $B$ , debemos tener  $|e| = |B|$ .

Exactamente  $|A|$  vértices de  $h$  son designados como vértices externos y numerados como  $1, \dots, |A|$ . Los otros vértices en  $h$  son llamados vértices internos.

Para cada regla de producción  $R_i : A \rightarrow h$ , denotaremos  $I(R_i)$  para referirnos a  $A$  (cabeza de la regla de producción, es decir parte izquierda de  $(R_i)$  y  $D(R_i)$  para referirnos a  $h$  (cuerpo de la regla de producción, es decir parte derecha de  $R_i$ )

El subhipergrafo es análogo a  $D(R_i)$  de una *GLC*. (Joshi y Schabes, 1997), este subhipergrafo se usa durante el proceso de derivación para reemplazarlo por una hiperarista, la cual está etiquetada con un elemento no terminal  $N$  en el hipergrafo, este reemplazo se lleva a cabo a través de los nodos externos ( $\bullet$ ) y la raíz, los cuales especifican como unir el subhipergrafo durante el proceso de derivación.

Los nodos externos se definen como una lista ordenada  $X \in V^*$  de nodos diferentes. El nodo raíz se define como un vértice designado como la raíz del hiperárbol y cada arista es dirigida desde esta raíz.

La Figur 2 inciso b) muestra el reemplazo de la hiperarista  $Y$  por el subhipergrafo  $D(R_3)$  de la regla de producción  $R_3 : A_3 \rightarrow h$  de la Figura2 inciso a).

### 3 Proceso de derivación

El proceso de derivación inicia con el símbolo inicial  $S$ , y de forma repetida se elige un símbolo  $A$  no terminal para ser reemplazado por el hipergrafo  $D(R_i)$ , usando una regla de producción  $R_i : A_i \rightarrow h$ . A su vez  $D(R_i)$  puede tener otras hiperaristas con etiquetas no terminales, de tal manera que el proceso continúa hasta que todas las hiperaristas solo tengan etiquetas terminales (Aguinaga et al., 2018).

**Definición 4** *Un hipergrafo  $H = (V, E = (e_i)_{i \in I})$  (Bretto, 2013) es isomorfo a un hipergrafo  $H' = (V', E' = (e'_j)_{j \in J})$ , si existe una biyección:*

$$f : V \rightarrow V'$$

y una biyección  $\pi : I \rightarrow J$ , la cual induce una biyección  $g : E \rightarrow E'$

tal que  $g(e_i) = e'_{\pi(i)}$ , para todo  $e_i \in E$ .

La pareja  $(f; g)$  es llamada un isomorfismo de hipergrafos.

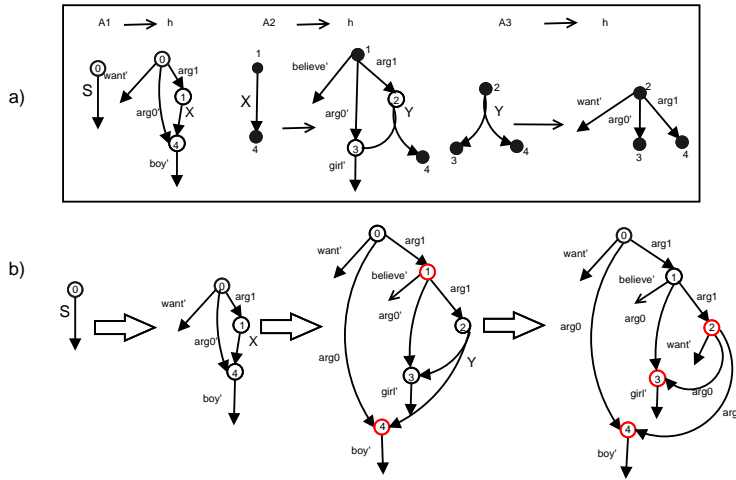
**Definición 5** *Sea  $G$  una HRG y  $p = (A \rightarrow R)$  una regla de producción de  $G$  (Aguinaga et al., 2018). La relación  $H_i \Rightarrow H^*$  ( $H^*$  es derivada de  $H_i$  en un solo paso) se define de la siguiente forma.  $H_i$  debe tener una hiperarista  $e$  etiquetada con  $A$ ; sea  $v_1, \dots, v_k$  vértices conectados de  $A$ . Sean  $u_1, \dots, u_k$  los vértices externos de  $R$ . Entonces  $H^*$  es el grafo formado después de haber removido  $e$  de  $H_i$ , haciendo una copia isomorfa de  $R$ , e identificando  $v_i$  con las copias de  $u_i$  para cada  $i = 1, \dots, k$ . Sea  $\overset{*}{\Rightarrow}$  la cerradura reflexiva, transitiva de  $\Rightarrow$ . Entonces decimos que  $G$  genera un hipergrafo  $H$  si hay una producción  $S \rightarrow R$  y  $R \overset{*}{\Rightarrow} H$  y  $H$  solamente tiene hiperaristas etiquetadas con símbolos terminales.*

La Figura 2 muestra un ejemplo del proceso de derivación a partir de la HRG  $G$  para el hipergrafo de entrada, el cual representa "The boy wants the girl to believe that he wants her".

Inicialmente, se aplica la regla  $R_1 : A_1 \rightarrow h$  para reemplazar el símbolo inicial  $S$  por el subhipergrafo  $D(R_i)$  posteriormente, en el hipergrafo resultante y en base a la

regla  $R_2 : A_2 \rightarrow h$  se reemplaza la hiperarista  $X$  por el subhipergrafo  $D(R_2)$ , de tal forma que se realice una copia isomorfa de  $D(R_2)$  sobre el hipergrafo, considerando como puntos de unión el nodo raíz ( $\bullet$ ) 1 y el nodo externo (4).

Finalmente, la hiperarista  $Y$  se reemplaza por el subhipergrafo  $D(R_3)$ , de tal forma que se realiza una copia isomorfa de  $D(R_3)$  sobre el hipergrafo, considerando como puntos de unión el nodo raíz ( $\bullet$ ) 2 y los nodos externos (3 y 4). En el hipergrafo resultante se puede observar que todas las hiperaristas están etiquetadas solamente con símbolos terminales y por lo tanto se concluye que el hipergrafo de entrada sí se deriva a partir de la HRG de entrada.



**Fig. 2.** a) Reglas de producción para la HRG  $G$ . b) Derivación a partir de  $G$  para el hipergrafo que representa el significado "The boy wants the girl to believe that he wants her" (Referenciada de Chiang et al. (2013))

Un lenguaje de hipergrafos es el conjunto de todos los hipergrafos  $H$ , los cuales contienen solamente hiperaristas terminales y que pueden ser derivados a partir del símbolo  $S$  en un número finito de pasos  $S \xRightarrow{*} H$  (Minas, 2000).

Si  $lab_H(e) = A$ , entonces se dice que  $H$  es el control inducido por  $A$  y se denota por  $A^\bullet$  (Rozenberg y Welzl, 1986).

**Definición 6** El lenguaje de hipergrafo  $L(HRG)$  (Rozenberg y Welzl, 1986), (Drewes et al., 1997) generado por HRG es  $L(HRG)$ , donde para todo  $A \in N$ ,  $L_A(HRG)$  consiste de todos los hipergrafos en  $H_T$  derivables de  $A^\bullet$  aplicando producciones de  $P$ .

$$L_A(HRG) = \{H \in H_T | A^\bullet \xRightarrow{*} H\}$$

**Definición 7** El grado o cardinalidad de un nodo  $x$  es el número de hiperaristas incidentes al nodo  $x$  y se denota por  $g(x)$ .



**Definición 8** El grado de salida de un nodo  $x$  es el número de hiperaristas que tiene al nodo  $x$  como nodo inicial y se denota por  $g^-(x)$ .

---

**Algorithm 1**  $BuscaSub([H], [R_i], n_H) \rightarrow [sub]$

---

- 1:  $g^-(n_H) = g^-(sub) = [sub]$
  - 2:  $h_l < -[R_i, D(R_i)], H_s < -[H]$
  - 3:  $h_l = H_s$
  - 4:  $add([sub], H_s)$
  - 5:  $BuscaSub(H, [R_i : R_{is}], n_H) = BuscaSub(H, R_{is}, n_H)$
- 

## 4 Proceso de Análisis

En esta sección se muestra la descripción del algoritmo para el análisis de las *HRGs*, a través de un hipergrafo de entrada  $H$ .

### 4.1 Analizador

Dados  $H$  un hipergrafo y  $G$  una *HRG*. La función *Analizador()* identifica subhipergrafos  $sub_i$  en el hipergrafo  $H$ , en base a las reglas de la  $G$ , de tal forma que se repite el siguiente proceso.

---

**Algorithm 2**  $BuscaReglas([H], [R_i], n_H) \rightarrow [R_i]$

---

- 1:  $BuscaReglas([H], [], n_H) = false$
  - 2:  $BuscaReglas(H, [R_i : R_{is}], n_H) = BuscaSub(H, R_i, n_H)$
  - 3:  $BuscaReglas(H, [R_i : R_{is}], n_H) = BuscaReglas(H, R_{is}, n_H)$
- 

Inicialmente, la función *BuscaReglas* identifica el subhipergrafo  $D(R_i)$  para el símbolo inicial  $R_i = S$  de  $G$  y a través de la función *BuscaSub* va construyendo el subhipergrafo  $sub_i$  para la parte  $D(R_i)$  isomorfo en  $H$ .

---

**Algorithm 3**  $Analizador([H], [G]) \rightarrow Bool$

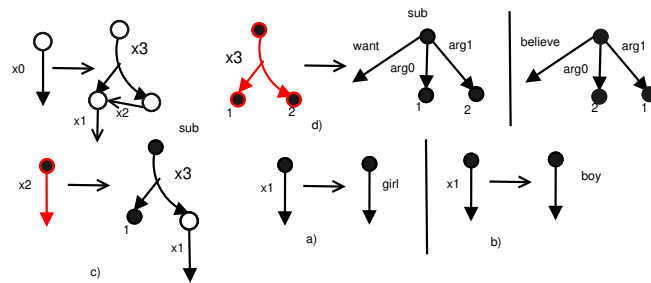
---

- 1:  $Analizador([], G) = true$
  - 2:  $Analizador(H, []) = false$
  - 3:  $Analizador(H, G) = Analizador(H \setminus BuscaReglas(H, R_i, n_{H_{sig}}), G)$
- 

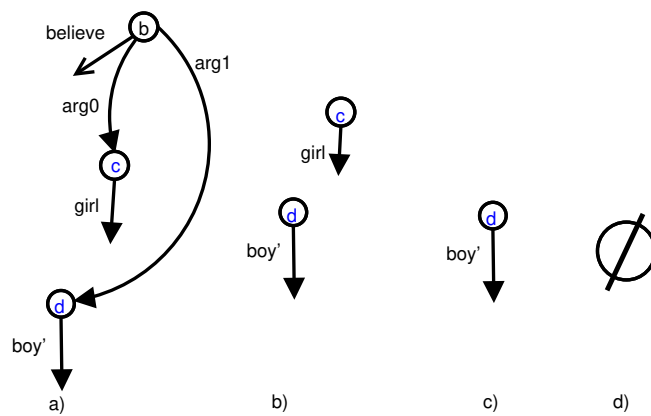
Enseguida, si la función *Analizador* recibe el subhipergrafo  $sub_i$  isomorfo en  $H$  lo elimina del hipergrafo  $H$  y avanza a la siguiente raíz ( $n_{H_{sig}}$ ) del hipergrafo  $H$ , es

decir al siguiente subhipergrafo y repite todo el proceso anterior para cada elemento  $N$  de  $R_i$ , mientras el hipergrafo  $H$  sea diferente de  $\emptyset$  y el subhipergrafo de la parte  $D(R_i)$  contenga elementos  $N$  o hasta haber revisado todas las reglas de  $G$ , es decir  $G = \emptyset$ .

Finalmente, en caso de no haber formado el subhipergrafo  $sub_i$  y siendo  $G \neq \emptyset$ , la función *BuscaReglas* busca una nueva regla  $R_i$  y repite todo el proceso anterior. En caso de que no haya más reglas para  $N$ , la función *BuscaReglas* devuelve el valor *false* para indicar que el proceso de análisis ya no puede continuar, por lo que la función *Analizador* determina que el hipergrafo  $H$  no es generada por  $L(G)$ . En caso contrario determina que  $H$  si es generado por  $L(G)$ .



**Fig. 3.** Reglas de producción para la HRG  
(Referenciada de Peng et al. (2015))



**Fig. 4.** a)  $H - sub_1$ , b)  $H - sub_3$ , c)  $H - sub_4$ , d)  $H - sub_5$

## 4.2 Ejemplo

Ahora, mostramos un ejemplo para verificar si el hipergrafo de entrada  $H$  de la Figura 2 inciso b) es generado por  $L(G)$  de la Figura 3.

La función *BuscaReglas* inicialmente, busca la regla correspondiente al elemento inicial  $R_i = x_0$  de la parte  $D(R_i)$  y debido a que esta contiene los elementos  $N = \{x_1, x_2, x_3\}$  como se observa en la Figura 3, enseguida *BuscaReglas* busca una regla para cada uno de los  $N$ , de tal forma que inicialmente, para la primera opción de la regla  $x_3$  la función *BuscaSub* construye el subhipergrafo  $sub_1$  de la parte  $D(R_{x_3})$  isomorfo en  $H$  como se observa en la Figura 3 inciso d), enseguida la función *Analizador* elimina el subhipergrafo  $sub_1$  del hipergrafo  $H$  como se observa en la Figura 4 inciso a) y se desplaza a la siguiente raíz  $n_{H_{sig}} = b$ .

Ahora, para la regla  $x_2$  la función *BuscaReglas* identifica el subhipergrafo  $sub_2$  de la parte  $D(R_{x_2})$ , el cual tiene a los elementos  $N = \{x_3, x_1\}$ , así que primero identifica el subhipergrafo  $sub_3$  de la parte  $D(R_{x_3})$  para la segunda opción de  $x_3$ , enseguida la función *BuscaSub* construye el subhipergrafo de la parte  $D(R_{x_3})$  isomorfo en  $H$  como se observa en la Figura 3 inciso d), enseguida la función *Analizador* elimina el subhipergrafo  $sub_3$  del hipergrafo  $H$  como se observa en la Figura 4 inciso b) y se desplaza a la siguiente raíz  $n_{H_{sig}} = c$ .

Posteriormente, la función *BuscaReglas* identifica el subhipergrafo  $sub_4$  de la parte  $D(R_{x_1})$  para la segunda opción de la regla  $x_1 \in x_2$ , enseguida la función *BuscaSub* construye el subhipergrafo de la parte  $D(R_{x_1})$  isomorfo en  $H$  como se observa en la Figura 3 inciso b), enseguida la función *Analizador* elimina el subhipergrafo  $sub_4$  del hipergrafo  $H$  como se observa en la Figura 4 inciso c) y se desplaza a la siguiente raíz  $n_{H_{sig}} = d$ .

Finalmente, la función *BuscaReglas* identifica el subhipergrafo  $sub_5$  de la parte  $D(R_{x_1})$  para la primera opción de la regla  $x_1 \in x_0$ , enseguida la función *BuscaSub* construye el subhipergrafo de la parte  $D(R_{x_1})$  isomorfo en  $H$  como se observa en la Figura 3 inciso a), finalmente la función *Analizador* elimina el subhipergrafo  $sub_5$  del hipergrafo  $H$  como se observa en la Figura 4 inciso d) y debido a que  $H = \emptyset$  la función *Analizador* concluye que el hipergrafo  $H$  si es generado por el lenguaje generado por  $G$ .

## 5 Conclusiones

La solución que se presenta no incrementa la complejidad, sin embargo se ve una disminución en los valores de las constantes involucradas, además es un método simple lo que facilita su implementación y a diferencia del método que presenta (Chiang et al., 2013) no se requiere del cálculo de una buena descomposición arbórea.

Como trabajo a futuro, el siguiente paso es desarrollar una versión parametrizada, en la cual se lleve a cabo un análisis de los diferentes parámetros para los hipergrafos asociados a las gramáticas *HRGs* tales como, el número de reglas, ancho arbóreo acotado de tal forma que permitan extender la clase de *HRGs* tratable. Así como determinar técnicas de descomposición arbórea para las gramáticas *HRGs* que garanticen algoritmos eficientes para la clase extendida.

## Referencias

- Aguinaga, S., Chiang, D., y Weninger, T. (2018). "Learning hyperedge replacement grammars for graph generation". *CoRR*, *abs/1802.08068*. Recuperado de: <http://arxiv.org/abs/1802.08068>.
- Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., ... Schneider, N. (2013). "Abstract meaning representation for sembanking". In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*. pp. 178-186. Association for Computational Linguistics. Recuperado de: <http://aclweb.org/anthology/W13-2322>.
- Bretto, A. (2013). "*Hypergraph theory: An introduction*". Springer Publishing Company, Incorporated.
- Chiang, D., Andreas, J., Bauer, D., Hermann, K. M., Jones, B., y Knight, K. (2013). "Parsing graphs with hyperedge replacement grammars". In *Proceedings of the 51st annual meeting of the association for computational linguistics. vol. 1: Long papers* pp. 924-932. Association for Computational Linguistics. Recuperado de: <http://aclweb.org/anthology/P13-1091>.
- Drewes, F., Kreowski, H.-J., y Habel, A. (1997). "Handbook of graph grammars and computing by graph transformation". In G. Rozenberg (Ed.), pp. 95-162. River Edge, NJ, USA: World Scientific Publishing Co., Inc. Recuperado de: <http://dl.acm.org/citation.cfm?id=278918.278927>.
- Flanigan, J., Thomson, S., Carbonell, J., Dyer, C., y Smith, N. A. (2014). "A discriminative graph-based parser for the abstract meaning representation". In *Proceedings of the 52nd annual meeting of the association for computational linguistics vol. 1: Long papers* pp. 1426-1436. Association for Computational Linguistics. Recuperado de: <http://aclweb.org/anthology/P14-1134>. doi: <https://doi.org/10.3115/v1/P14-1134>
- Groschwitz, J., Koller, A., y Teichmann, C. (2015). "Graph parsing with s-graph grammars". In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing of the asian federation of natural language processing, ACL 2015, july 26-31, 2015, beijing, china, vol. 1: Long papers* pp. 1481-1490. The Association for Computer Linguistics. Recuperado de: <http://aclweb.org/anthology/P/P15/P15-1143.pdf>.
- Habel. (1991). "*Hyperedge replacement: Grammars and languages*" (1st ed.). United States of America: University of Bremen.
- Joshi, A. K., y Schabes, Y. (1997). "Tree-adjoining grammars". In G. Rozenberg y A. Salomaa (Eds.), *Handbook of formal languages, vol. 3*, pp. 69-124. Berlin, New York: Springer. Recuperado de: <http://www.cs.cmu.edu/afs/cs.cmu.edu/project/cmt-55/lti/Courses/722/Spring-04/readings/Joshi-Schabes-TAGs.pdf>.
- Lautemann, C. (1990, Apr 01). "The complexity of graph languages generated by hyperedge replacement". *Acta Informatica, vol. 27, núm. 5*, pp. 399-421. Recuperado de: <https://doi.org/10.1007/BF00289017>. doi: <https://doi.org/10.1007/BF00289017>

- Minas, M. (2000). "Hypergraphs as a uniform diagram representation model". In *Selected papers from the 6th international workshop on theory and application of graph transformations* pp. 281-295. London, UK, UK: Springer-Verlag. Recuperado de: <http://dl.acm.org/citation.cfm?id=645872.671543>.
- Peng, X., Song, L., y Gildea, D. (2015). "A synchronous hyperedge replacement grammar based approach for amr parsing". In *Proceedings of the nineteenth conference on computational natural language learning* pp. 32-41. Association for Computational Linguistics. Recuperado de: <http://aclweb.org/anthology/K15-1004>. doi: <https://doi.org/10.18653/v1/K15-1004>
- Rozenberg, G., y Welzl, E. (1986, 04). "Boundary nlc graph grammars basic definitions, normal forms, and complexity". *Information and Control*, vol. 69, pp. 136-167. doi: [https://doi.org/10.1016/S0019-9958\(86\)80045-6](https://doi.org/10.1016/S0019-9958(86)80045-6)

# Capítulo 16

## Preprocesamiento de imágenes para la autenticación de personas

Aida A. Aparicio-Arroyo, Ivan Olmos-Pineda, J. Arturo Olvera-López

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

aida.aparicio@alumno.buap.mx, {iolmos,aolvera}@cs.buap.mx

**Resumen.** En el presente trabajo se muestra, en primer lugar, el diagrama general del proceso de autenticación de personas. El cual generalmente consta de 4 etapas: detección de rostros, preprocesamiento, extracción de características y la clasificación. En seguida, se presentan algunos trabajos relacionados a la etapa de preprocesamiento, enfocándose a la corrección de pose. Luego se sugiere una metodología del proceso de autenticación, incluyendo la etapa de corrección de pose y la extracción de 3 bloques de características: las basadas en pigmentación de la piel, las basadas en la textura de la piel y las basadas en geometría; esto con el objetivo de aumentar la precisión de la autenticación y de igual manera, mantener un tiempo de respuesta corto, comparándolo con el tiempo que una persona se puede tardar en recordar a otra persona. Y posteriormente, se exponen algunos experimentos realizados para el proceso de autenticación de personas.

**Palabras Clave:** Autenticación de personas, visión por computadora, extracción de características.

### 1 Introducción

En la actualidad, es común escuchar sobre la inteligencia artificial, también es común hablar sobre los robots y los humanoides, pero tal vez no es común hablar, de cómo es posible que un robot pueda realizar ciertas tareas como: entablar un diálogo, reconocer a las personas, desplazarse de un lugar a otro esquivando ciertos obstáculos, o que incluso, un robot pueda realizar tareas difíciles para el ser humano como: desactivar bombas, operaciones quirúrgicas muy peligrosas, entre otras.

Para que el robot pueda desarrollar estas actividades, un conjunto de áreas de investigación entran en juego como: la robótica, la visión por computadora, la inteligencia artificial, la lingüística, entre otras.

En el presente trabajo nos enfocaremos en el área de visión por computadora, con el objetivo de presentar una metodología que sea capaz de realizar la tarea de autenticación de personas dentro de la interacción humano-robot. Esta tarea se debe llevar a cabo sin importar la posición en la que se encuentre la persona con respecto al robot y, de igual manera, esta metodología debe de tener un tiempo corto de respuesta, ya que se piensa comparar con el tiempo en que una persona puede tardar en autenticar o recordar a otra persona.

Actualmente, existen sistemas de vigilancia que son capaces de autenticar personas, pero estos sistemas trabajan bajo las condiciones de que la persona esté de frente a la cámara (sistema) y a cierta distancia, algunos ejemplos de este tipo de sistema son los utilizados en los aeropuertos de Estados Unidos, Singapur y China para agilizar el proceso de abordaje Orona (2019). Estos sistemas aeroportuarios, utilizan sistemas biométricos, es decir, extraen características faciales como la geometría de los ojos, nariz y la boca, estas características son incluidas en un descriptor, el cual es utilizado para la clasificación (Chowdhury et al., 2016).

En la figura 1, se muestra a grandes rasgos el proceso de autenticación de personas utilizando los rasgos faciales.

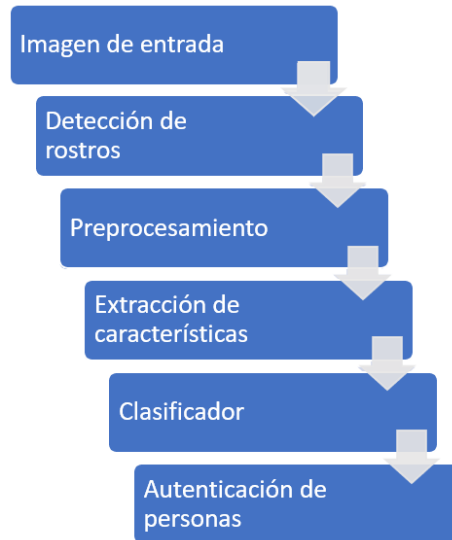


Figura 1. Proceso general para la autenticación de personas

## 2 Trabajos relacionados

Con base en el proceso general para la autenticación de personas mostrado en la figura 1, en esta sección se presentarán algunos trabajos relacionados a la etapa de

preprocesamiento, la cual puede consistir en corregir la imagen (a nivel de valores de píxeles, contraste, iluminación, etc.), aplicar filtros digitales, determinar la posición en la que se encuentra el rostro de la persona, hacer rotación o traslación de la imagen del rostro, entre otros procesos. Esta etapa, basándose en artículos de investigación, es necesaria para la extracción de características, para así obtener resultados satisfactorios en el proceso de autenticación.

Dentro de la etapa de preprocesamiento, una de las modificaciones realizadas es la corrección de la pose del rostro. Autores como Beymer (1993) y González-Jiménez y Alba-Castro (2007), implementan plantillas o mallas, las cuales son usadas para estimar la postura de la cabeza, hacen una correlación entre una plantilla base y la plantilla correspondiente a la imagen de entrada.

Otros autores como Rowley et al (1998) y Bormann et al (2013), utilizan 2 clasificadores para poder determinar la postura en la que se encuentra el rostro de la persona. Un clasificador se utiliza para extraer características de los rostros en diferentes posturas y el otro clasificador, es entrenado con la capacidad de estimar la orientación del rostro utilizando las características anteriormente extraídas.

Dentro de la literatura, aparte de utilizar plantillas o mallas, o clasificadores para determinar la orientación del rostro, hay autores que utilizan modelos en 3D generados a partir de imágenes en 2D. Como es el caso de Kong y Mbouna (2015), Goswami et al (2016) y Samad e Iftekharuddin (2016). Realizan proyecciones entre ambos planos, a través de una matriz de proyección, la cual relaciona las coordenadas de un punto en el espacio 3D con su proyección en el plano 2D. También, trabajando con los modelos en 3D, se obtienen mapas de profundidad, los cuales aportan información sobre la profundidad del rostro y la posición en la que se encuentra. De igual manera, se proponen las curvas faciales en 3D. A través de un marco matemático, se mide la invarianza de la pose, para posteriormente, extraer las características.

Estos son algunos de los procesos para la detección de orientación, los cuales sirven para una posterior corrección de pose del rostro. Estos procesos tienen sus pros y sus contras. Las mallas pueden ser ampliadas a un conjunto de diferentes poses, pero necesitan un método de interpolación para estimar la ubicación de la cabeza. Los clasificadores emplean algoritmos de entrenamiento que ignoran la variación de apariencia, pero puede ser tardado el entrenamiento de los clasificadores. Con respecto a los modelos en 3D, no necesita algún preprocesamiento la imagen y la generación de los modelos pueden ser rápidos, pero se necesita generar un modelo para cada orientación detectada.

### **3 Metodología propuesta**

En la figura 2, se muestra la metodología que se sugiere, la cual se propone con el fin de aumentar la precisión dentro del proceso de autenticación de personas y que, además, este proceso sea invariante a la posición en la que se encuentre la persona, es decir, que no



importe si la persona está de frente o de perfil con respecto al robot con el que está interactuando.

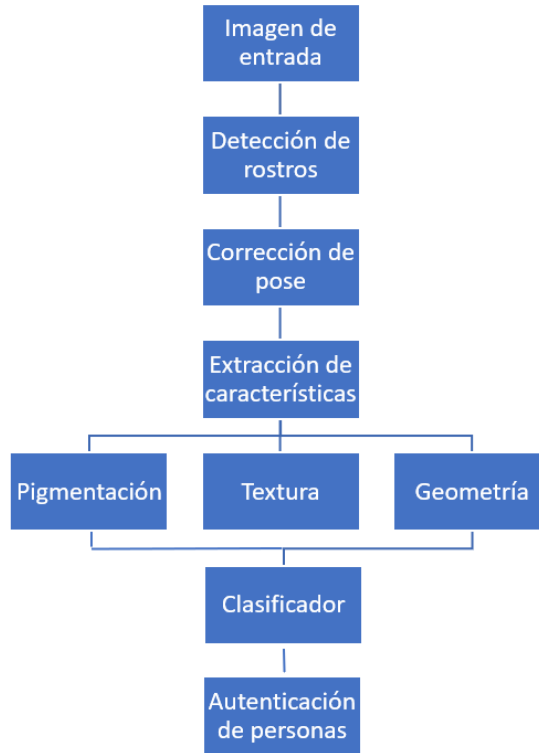


Figura 2. Diagrama general de la metodología propuesta

Se tiene una imagen de entrada, la cual es capturada por una cámara colocada en un robot. La primera etapa, consiste en la detección de los rostros de las diferentes personas que se encuentran interactuando con el robot. Dentro de la literatura, uno de los algoritmos más implementados para esta etapa es el de Viola-Jones, pero este algoritmo tiene buenos resultados debido a que las personas se encuentran totalmente de frente a la cámara, pero para el caso de investigación, las personas pueden estar volteando hacia cualquier lado, por tal motivo se utilizará una red neuronal convolucional, la cual será entrenada para detectar los rostros sin importar hacia dónde está mirando las personas.

Después de haber detectado los rostros dentro de la escena (salón de clases, laboratorio, etc.), se debe determinar la orientación del rostro, es decir, determinar hacia dónde está posicionada la cara (frente, perfil derecho, perfil izquierdo), esto con el objetivo de corregir la pose del rostro.

Como tercera etapa, se encuentra la extracción de características. Ya después de haber corregido la pose del rostro y haber escalado todas las imágenes de los rostros, es momento de extraer las características más relevantes de la cara. Se pueden dividir en 3 áreas: características basadas en pigmentación (color), características basadas en textura y características basadas en geometría (forma del rostro, distancia de zonas de interés, etc.).

En cada una de las áreas se extraerán características diferentes, las cuales posteriormente, se almacenarán en un vector característico, el cual servirá para entrenar un clasificador y así obtener un modelo, que al final será evaluado en el proceso de autenticación de personas.

#### 4 Resultados preliminares

Se ha generado una base de datos de 300 imágenes de 10 personas diferentes (5 mujeres, 5 hombres), en promedio 30 imágenes por persona. En la figura 3, se muestran algunas de las imágenes obtenidas. Estas imágenes fueron tomadas por la misma persona a diferentes distancias y circunstancias.



Figura 3. Algunos ejemplos de la base de imágenes capturadas

#### 4.1 Detección de rostros

Con base en el diagrama presentado en la sección anterior, la primera etapa es la detección de rostros. Esta etapa se ha realizado utilizando una red neuronal convolucional (CNN, por sus siglas en inglés) ya entrenada. En la figura 4, se muestran algunas de las imágenes resultantes después de la implementación del programa Mask R-CNN. El porcentaje de precisión abarcaba desde el 80% hasta el 99 %.



Figura 4. Detección de personas e imagen segmentada

#### 4.2 Corrección de pose y escalamiento

Dentro de la literatura, se encontraron algunos algoritmos utilizados para la corrección de pose en el eje z. Estos algoritmos se basan en ubicar el centro de los ojos, los cuales se tomarán como referencia para determinar el ángulo en el que se encuentra la imagen de entrada. Acto seguido, se hace la corrección a través de una matriz de proyección y, para la parte del escalamiento, se puede realizar mediante transformaciones lineales. En la figura 5, se observan algunos de los resultados obtenidos de la utilización de un algoritmo para la corrección de pose en el eje z y de escalamiento.



*Figura 5.* Corrección de pose en el eje z y escalamiento

De las 300 imágenes con las que se probó el algoritmo utilizado, solamente 275 imágenes (91.66 %) fueron corregidas, mientras que 25 imágenes (8.34 %) no se obtuvieron resultados. Analizando algunos casos donde no hubo una corrección, los rostros de las personas tenían algún tipo de sombra, ya sea por el momento en el que se tomó la foto, o por tener el rostro en cierta inclinación.

#### **4.3 Landmarks de zonas de interés**

Una parte para la extracción de características es la identificación de las zonas de interés como: las cejas, los ojos, la nariz y la boca. En este caso, se utiliza una plantilla para la localización de las diferentes zonas de interés, esta plantilla contiene las coordenadas de donde se localizan estas zonas. En la figura 6, se pueden observar algunos de los ejemplos de las zonas de interés que se pueden localizar dentro de las imágenes.



Figura 6. Localización de zonas de interés

En la tabla 1, se muestran los resultados cuantitativos de la detección de los landmarks de las diferentes zonas de interés. Cabe mencionar que, estos resultados se obtuvieron aplicando el algoritmo en las 275 imágenes ya corregidas en la pose y el escalamiento.

Tabla 1

*Detección de zonas de interés*

Resultado\Zona	Boca	Ceja der.	Ceja izq.	Ojo der.	Ojo izq.	Nariz	Mentón
Detectado	258	258	258	258	258	258	254
No detectado	17	17	17	17	17	17	21

#### 4.4 Extracción de características

Como se menciona en el capítulo anterior, se analizan tres áreas: pigmentación en zonas de piel, textura en la piel y geometría. Las imágenes analizadas son donde el rostro

se encuentra de frente al dispositivo de captura. Estas imágenes de prueba son de 4 personas diferentes (2 mujeres y 2 hombres), las cuales fueron tomadas en diferentes circunstancias y distancias. Por cada persona se cuenta con una muestra de 17 imágenes. Para cada una de las áreas a analizar, se hizo un preprocesamiento diferente como: ecualización del histograma, aplicación de filtro para aumentar el contraste, corrección de pose en el eje z, escalamiento, detección de bordes, entre otras. Ya después de hacer el preprocesamiento, es momento de extraer las características para cada una de las áreas a analizar.

En la tabla 2, se en listan las diferentes características que se han extraído hasta el momento. Dichas características se han incluido en un descriptor, el cual se ha introducido en un clasificador SVM, obteniendo un 69% de precisión.

Tabla 2

*Características extraídas*

Segmento	Características
Pigmentación	Desviación, varianza, entropía, correlación
Textura	GLCM, Fourier, Gabor
Geometría	Ancho del rostro, distancias de zonas de interés, distancia entre zonas de interés

A pesar de haber corregido las imágenes con relación a la postura del rostro, se obtuvo un porcentaje bajo de precisión, este comparado con algunos trabajos relacionados que van arriba del 88%. Se observa que se necesita modificar esa etapa (corrección de pose), ya sea cambiando algunos parámetros del algoritmo o usar otro proceso de estimación y corrección de pose, con el objetivo de aumentar el porcentaje de precisión, pero siempre tomando en cuenta el tiempo de ejecución.

## 5 Conclusiones

Los experimentos realizados se hicieron con las imágenes de la base de datos que se ha generado hasta el momento, pero cabe mencionar que, en su mayoría son imágenes de frente. Al momento de la extracción de características, las imágenes no tuvieron un preprocesamiento adecuado para cada una de las áreas, por ejemplo, no se aplicaron filtros para resaltar el contraste en las imágenes, o para aumentar la iluminación, o para eliminar el ruido de las imágenes, entre otras cosas.

Como trabajo a futuro, se propone corregir la parte de detección de orientación, esto para poder determinar la orientación del rostro y así poder extraer las características correspondientes a cada posición (frente, perfil). De igual manera, se plantea aplicar diferentes pasos de preprocesamiento a las imágenes, esto con el fin de poder tener las imágenes procesadas para la etapa de extracción de características.

## Referencias

- Beymer, D. (1994). "Face recognition under varying pose", *Proceedings of 23rd Image Understanding Workshop*, pp. 837-842.
- Bormann, R., Zwölfer, T., Fischer, J., Hampp, J., y Hägele, M. (2013). "Person recognition for service robotics applications", In *Humanoid Robots (Humanoids), 2013 13th IEEE/RSJ International Conference on*, pp. 260–267.
- Chowdhury, M., Gao, J., e Islam, R. (2016). "Fuzzy rule based approach for face and facial feature extraction in biometric authentication", In *Image and Vision Computing New Zealand (IVCNZ), 2016 International Conference on*, pp. 1–5.
- González-Jiménez, D. y Alba-Castro, J. L. (2007). "Toward pose-invariant 2-d face recognition through point distribution models and facial symmetry", *IEEE Transactions on Information Forensics and Security*, vol. 2, pp. 413–429.
- Goswami, K., Lee, J.-H., Jang, K.-S., Kim, B.-G., y Kwon, K.-K. (2016). "Entropy difference-based early skip detection technique for high-efficiency video coding", *Journal of Real-Time Image Processing*, vol. 12, pp. 237–245.
- Kong, S. G., y Mbouna, R. O. (2015). "Head pose estimation from a 2d face image using 3d face morphing with depth parameters", *IEEE Transactions on Image Processing*, vol. 24, pp. 1801–1808.
- Orona, K. (2019). *Aerolíneas usan reconocimiento facial como pase de abordar*. Recuperado de <https://www.eluniversal.com.mx/destinos/aerolineas-usan-reconocimiento-facial-como-pase-de-abordar>
- Rowley, H. A., Baluja, S., y Kanade, T. (1998). "Rotation invariant neural network-based face detection", In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 38–44.
- Samad, M. D., e Iftikharuddin, K. M. (2016). "Frenet frame-based generalized space curve representation for pose-invariant classification and recognition of 3-d face", *IEEE Transactions on Human-Machine Systems*, vol. 46, pp. 522–533.

# Capítulo 17

## Análisis de técnicas para el Reconocimiento Automático de Lengua de Señas

Daniel Sánchez-Ruiz<sup>1</sup>, J. Arturo Olvera-López<sup>1</sup>, Ivan Olmos-Pineda<sup>1</sup>, Soraia Prietch<sup>2,3</sup>

<sup>1</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

<sup>2</sup> Universidade Federal de Mato Grosso  
Facultad de Ciencias de la Computación

<sup>3</sup> Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Electrónica

daniel.sanchez.ruiz@protonmail.com, {aolvera, iolmos}@cs.buap.mx,  
soraia@uftm.br

**Resumen.** En la actualidad los avances tecnológicos nos ayudan a tener mejores interacciones, incluso permiten la comunicación entre comunidades que lo hacen de forma distinta o limitada como lo son las personas sordas, que ocupan como lengua algún lenguaje de señas y las personas que basan su comunicación en una lengua hablada; esto se ha logrado gracias a los avances en el área del procesamiento de lenguaje de señas, donde se definen tres grandes áreas: el reconocimiento, la generación y la traducción automática de las distintas lenguas de señas. En este trabajo se realiza una revisión sobre el reconocimiento automático de lengua de señas, en él se identifican los principales sistemas, las características que lo componen, las bases de datos que han sido ocupadas como referencia y se discuten posibles áreas de oportunidad para investigaciones futuras.

**Palabras Clave:** Visión Computacional, Reconocimiento de Lengua de Señas, Reconocimiento de Patrones, Procesamiento de Imágenes Digitales

### 1 Introducción

Actualmente se calcula que 466 millones de personas tienen alguna discapacidad de pérdida de audición en todo el mundo, lo cual representa alrededor del 5% de la población total, y se estima que para el año 2050 habrá 900 millones de personas con este tipo de discapacidad (WHO, 2020). Tener una discapacidad auditiva significa que una persona no es capaz de escuchar del mismo modo que lo hace una persona oyente (Umbral de



audición de 25 dB o más en ambas orejas) (WHO, 2020). Existen dos categorías principales de discapacidad auditiva: problemas de audición, que se refiere a personas con pérdida de audición media a severa (que generalmente suelen comunicarse de forma hablada y que mejoran su capacidad auditiva a través de implantes o dispositivos auditivos), la otra categoría es la de sordos, donde se tiene pérdida de audición de forma profunda, lo cual implica una audición extremadamente limitada o la supresión total de todo tipo de capacidad auditiva. Este último grupo de personas suele comunicarse a través de lenguas de señas (WHO, 2020).

De acuerdo con la Federación Mundial de Sordos, existen alrededor de 300 lenguas de señas en el mundo, y 70 millones de personas sordas que los usan (WDF, 2020). La comunidad de personas sordas creó una lengua que hace uso de señas gesticuladas a través de las manos y que se complementa con expresiones faciales y lenguaje corporal. Este lenguaje es un medio de comunicación con el cual buscan poder transmitir sus sentimientos, ideas y necesidades. Al igual que los lenguajes hablados, las lenguas de señas se componen de distintas reglas gramaticales y vocabulario dependiendo de la región, inclusive en algunos casos hay lenguas con distintos componentes subregionales, por lo cual no se puede hablar de una lengua de señas universal.

Actualmente los avances tecnológicos en distintas áreas son remarcables, por lo cual ahora más que en ningún otro momento se busca mejorar la calidad de la vida de las personas. No obstante, en el caso de las tecnologías de la comunicación, la gran mayoría sólo tienen soporte para lenguaje hablado o escrito, excluyendo a las lenguas de señas y hay pocas personas que dominen alguna lengua de señas, por lo cual la comunidad sorda sufre de una barrera de comunicación con la mayoría de las personas (Bragg et al., 2019).

El procesamiento de lengua de señas ayudaría a romper estas barreras de comunicación para la comunidad sorda, esta área de procesamiento se refiere a las tareas de reconocimiento, generación y traducción de lengua de señas. El reconocimiento automático de lengua de señas es el primer paso en cualquier sistema de procesamiento de lengua de señas y es el tema de interés de este trabajo. El reconocimiento automático de lengua de señas es por si misma un área de investigación que involucra reconocimiento de patrones, visión por computador, procesamiento de lenguaje natural y lingüística. Su objetivo es construir métodos y algoritmos con el propósito de identificar señas pertenecientes a una lengua de señas y su significado correspondiente (Wadhawan y Kumar, 2019). Las lenguas de señas pueden dividirse en tres categorías principales para su reconocimiento: alfabetos y números, palabras y oraciones.

En este trabajo se presenta un revisión breve sobre el trabajo relacionado para el reconocimiento automático de lengua de señas que se ha propuesto en los últimos años, en particular se revisan los sistemas que se han propuesto, las características que han sido relevantes para la tarea del reconocimiento, los métodos y modelos de clasificación empleados en la fase de reconocimiento de patrones, las bases de datos que más se han ocupado, así como una discusión con los principales problemas y áreas de oportunidad que se han reportado. Esta revisión tiene como propósito fungir como un punto inicial en cualquier investigación referente al tema.

La organización del trabajo es de la siguiente forma: en el Capítulo 2 se abordan los distintos tipos de sistemas para el reconocimiento automático de lenguaje de señas que se han propuesto; el Capítulo 3 aborda los componentes que componen a los sistemas de reconocimiento automático de lenguaje de señas; el Capítulo 4 trata sobre algunas de las bases de datos que han sido ocupadas como referencia; en el Capítulo 5 se presenta una breve discusión sobre el contenido de los capítulos previos así como una revisión de los retos más importantes y posibles direcciones futuras en este campo de investigación; finalmente en el Capítulo 6 se listan las conclusiones del trabajo.

## 2 Sistemas de Reconocimiento Automático de Lenguaje de Señas

Un sistema de reconocimiento automático de lengua de señas puede ser clasificado en tres categorías con base a las señas que se busca reconocer: Deletreo con dedos, palabras aisladas y oraciones con señas continuas (Mohandes et al., 2014).

El deletreo de palabras es usado en situaciones donde palabras nuevas, nombres de personas, lugares o palabras no tienen una seña definida, por lo cual tienen que ser “deletreadas” por movimientos de manos (Kamal et al., 2019). En estos sistemas, las letras del alfabeto son presentadas de diversas formas o con distintos movimientos de dedos. En (Hu, 2018) se presenta una metodología que hace uso de información de profundidad (datos obtenidos del dispositivo Kinect) e intensidad para el reconocimiento de letras del alfabeto de la lengua de señas americano. Una segmentación con base a información de profundidad es propuesta y una vez con las regiones de interés se hace la extracción de características a partir del Histograma de Gradientes Orientadas (HOG), Patrones Locales Binarios (LBP) y momentos de Zernike, y finalmente un clasificador del tipo máquina de vectores de soporte (SVM), obteniendo buenos resultados. Por otro lado (Shi, Del Rio, Keane, Michaux, et al., 2019) introdujeron un enfoque para hacer el reconocimiento de deletreo de palabras *in the wild*, es decir ocupando videos recolectados de sitios web y no de una base de datos capturada siguiendo algún protocolo de adquisición, su método hace uso del detector de objetos Faster R-CNN, pero modificado para la detección de manos, una vez con la región de interés se alimenta una red neuronal convolucional recurrente con memoria de largo y corto plazo (LSTM), la cual da como salida la entrada a dos enfoques de reconocimiento de secuencias temporales, un decodificador LSTM basado en la atención y clasificación temporal conexionista (CTC); de acuerdo con los autores este el primer esfuerzo en esta área haciendo uso de los datos que ellos ocuparon. Posteriormente (Shi, Del Rio, Keane, Brentari, et al., 2019) desarrollaron una metodología que no se basa en la detección de las manos como en su trabajo previo, en este caso propusieron mecanismo de atención iterativa a través de operaciones de convoluciones, este modelo reduce el área de atención mientras de manera simultánea incrementa la resolución de la región de interés. Por otro lado (Tanwar et al., 2019) abordaron el problema con un enfoque progresivo respecto a aspectos relacionados al movimiento, analizando los gestos de la persona en cuestión. La segmentación de las

manos la realizan con un modelo gaussiano mixto, después una bolsa de palabras es ocupada para capturar las características referentes al movimiento, esto sirve como entrada a sus experimentos de clasificación donde ocuparon modelos de clasificación de SVM y Bosque Aleatorio. En general el problema de reconocimiento de deletreo de palabras se puede considerar como un problema básico de clasificación de imágenes, pues se suele trabajar con imágenes estáticas pertenecientes al alfabeto de señas, por esta razón esta área es menos estudiada respecto al reconocimiento de palabras aisladas y de oraciones con señas continuas.

En la categoría del reconocimiento de palabras aisladas, se analiza una secuencia de imágenes o señales de los movimientos de las manos capturadas a través de guantes que cuentan con sensores específicos, las cuales van a representar una seña. En esta categoría se reconocen alfabetos, números y palabras, ya que sólo un gesto se realiza en un momento dado. En un trabajo reciente (Lim et al., 2016) presentaron un enfoque basado en la eliminación de la cara y el fondo con técnicas básicas de procesamiento de imágenes digitales para preservar sólo la región de la mano. Posteriormente se hace el seguimiento de las manos y se calculan histogramas del flujo óptico en cada imagen en la secuencia en cuestión, estos histogramas se concatenan y funcionan como el vector de características conteniendo información de forma y tiempo de la región de interés, estos vectores se comparan con los generados en la etapa de entrenamiento y se calcula la diferencia, la menor será la tomada como la más probable. Ocupando de dos etapas principales: el seguimiento de la mano y la representación de la misma, en (Lim et al., 2019) propusieron una metodología para el reconocimiento aislado de lengua de señas, para el seguimiento de la mano hacen uso de la técnica de filtro de partículas que detecta el movimiento de las manos y junto con una red neuronal convolucional (CNN) entrenada con modelos de la mano, realizan la segmentación y para la representación se calcula un promedio de la región previamente segmentada y se genera una imagen que representa la energía en la secuencia de la seña. En (Ferreira et al., 2019) exploran el aprendizaje con datos multimodales, para ello extraen características con base al color y profundidad con base a datos adquiridos del dispositivo Kinect e información proveniente del dispositivo Leap, con estos datos se proponen diversos enfoques para su fusión en la etapa de clasificación. En (Jimenez et al., 2017) se introduce una metodología donde se reconoce el significado señas alfanuméricas, se hace énfasis en hacer una etapa de preprocesamiento donde entre otras cosas se unifica el tamaño de las manos de las personas a través de operaciones de escalamiento y puntos de referencia de una mano muestra, posteriormente se extraen características con una transformada de Haar pero en 3D, ya que se cuenta con información de profundidad dado que los datos fueron adquiridos con el dispositivo Kinect, finalmente un clasificador AdaBoost es ocupado. El reconocimiento de palabras aisladas es más útil que el del deletreo de palabras, y si bien la mayoría de los trabajos son basados en técnicas de visión, cada vez se ocupan enfoques que ocupan información de profundidad, donde dispositivos como el Kinect y Leap son cruciales para estos sistemas.

Finalmente, en la última categoría de reconocimiento de oraciones con señas continuas se busca reconocer el contenido de una conversación a través de videos. Esta categoría es la más relevante para situaciones de comunicación en las cuales se ven involucrados

individuos de la comunidad sorda en la vida real. Uno de los primeros trabajos en esta área lo presentan (Koller et al., 2015) un enfoque que hace uso de características multimodales en particular aquellas que se refieren a la forma, posición y movimiento de las manos, pero también empiezan a explorar expresiones faciales como características relevante, además de explorar esos datos, se detalla una base de datos para el reconocimiento de oraciones con señas continuas que se ha convertido en una referencia. Posteriormente (Koller et al., 2018) introdujeron una nueva metodología donde hacen uso de técnicas de aprendizaje profundo, algo que en las últimas investigaciones es cada vez más ocupado, en particular en su trabajo se introduce una CNN en un modelo oculto de Markov (HMM), esto se hace con el proposito de ocupar las habilidades discriminativas de las CNN pero también el modelado de secuencias de los HMM, el resultado de esa fase se procesa con un modelo bayesiano. En otro trabajo (Cui et al., 2017) exploran un enfoque similar donde igual ocupan CNN para extraer características espacio temporales, sin embargo en su caso ocupan una LSTM para el modelado de secuencias, es decir la clasificación de las señas, en el trabajo mencionan que obtienen mejores resultados a comparación de (Koller et al., 2018). Un acercamiento distinto al problema lo presentan (Camgoz et al., 2017) donde se emplean pequeñas sub redes que ellos detallan modelan conocimiento específico, en su caso relacionado a características espaciales y temporales, en cada subred se emplea un enfoque similar donde se hace uso de CNN y LSTM. Sin embargo, ellos agregaron en sus experimentos un enfoque de fusión donde consideraron datos de la región de la mano y del frame completo. En una investigación más reciente (Elakkiya y Selvamani, 2019) se ocupa un concepto similar al de las sub redes, solo que en este caso se definen como sub unidades, pero en este caso el modelado de cada sub unidad no hace uso de métodos de aprendizaje profundo, hacen uso del modelado espacial a través de operaciones geométricas y de posición con respecto a la región de las manos, en particular la posición también sirve para modelar el comportamiento temporal, con estas características definen un léxico sobre los datos y este sirve para la etapa de clasificación. Si bien la investigación sobre esta área está incrementando en los últimos años aun es limitada, sin embargo, esta área es la de más utilidad en las aplicaciones que se buscan crear para su uso diario en situaciones reales por lo cual es de vital importancia seguir sumando esfuerzos en presentar mejores metodologías con datos diversos que presenten situaciones realistas.

### **3 Características de los Sistemas de Reconocimiento Automático de Lengua de Señas**

Un sistema de reconocimiento automático de lengua de señas se compone de tres bloques básicos que a su vez pueden tener varios sub bloques: la adquisición de los datos, donde se seleccionan los dispositivos que serán ocupados para la captura; las características que se van a utilizar, donde suelen incluirse la fase de la segmentación y

los modelos de clasificación que se ocupan en la fase de reconocimiento de patrones (Kamal et al., 2019).

Generalmente para la adquisición de datos es a través de dispositivos de visión computacional o basados en sensores. Los enfoques basados en visión utilizan la adquisición de imágenes y videos para la tarea del reconocimiento. Típicamente se trabaja con dos tipos de dispositivos de visión, los que capturan imágenes en 2D, generalmente en formato RGB y los que capturan además información de profundidad y de trayectoria a través del dispositivo Kinect. Por otro lado, el enfoque basado en dispositivos con sensores, emplean guantes que cuentan con sensores o instrumentos de medición de datos que están adheridos a la mano. Estos sensores proveen información de ubicación, orientación, movimiento e inclinación de los dedos de la mano (Kamal et al., 2019) .

Después de la adquisición de los datos inicia la etapa del procesamiento, en primera instancia se define que tipo de características se van a ocupar, en (Zieren et al., 2006) se menciona que para cualquier lengua de señas hay que considerar dos tipos de características: las manuales y las no manuales. Las características manuales son las referentes a las manos, abarcando conceptos como la forma, la posición, la forma y la ubicación de estas y las características no manuales, que son aquellas que enfatizan el significado de las señas o que sirven para diferenciar señas similares, para ello se hace uso de expresiones faciales, postura de cuerpo, postura de cabeza, así como la forma de labios y boca. Recientemente también se ha identificado un componente que puede fungir como característica, los movimientos de epéntesis, estos movimientos son transiciones que se generan entre el fin de una seña y el inicio de otra, estos movimientos han servido para clasificar de forma errónea algunas señas en el pasado, por eso la correcta identificación de ellos se considera que ayudara a obtener mejores resultados (Ibrahim et al., 2019). Dependiendo de las características que se seleccionen se considerara un enfoque de segmentación de la región o regiones de interés, aquí se ocupan enfoques basados en el color de la piel, de regiones que presentan mayor movimiento o de algoritmos como Viola-Jones o CNN para identificar la cabeza o la postura del cuerpo, pero recientemente se han ocupado enfoques de aprendizaje profundo con las CNN donde ya no es necesario segmentar nada (Wadhawan y Kumar, 2019).

Finalmente con los vectores de características que se han generado se proceda a la etapa de clasificación, en esta etapa tradicionalmente se hacían uso de HMM para el modelado de secuencias, en el caso del deletreo de palabras se han ocupado SVM mayormente (Bragg et al., 2019; Kamal et al., 2019; Wadhawan y Kumar, 2019). Sin embargo, en trabajos recientes ha habido un incremento en el uso de enfoques de aprendizaje profundo, desde CNN, LSTM o modelos híbridos (Bragg et al., 2019; Kamal et al., 2019; Wadhawan y Kumar, 2019).

## 4 Conjuntos de Datos para Validación

En el reconocimiento automático de lengua de señas existen bases de datos de referencia que son tomadas en cuenta para comparar resultados en las investigaciones realizadas. Una de las primeras bases de datos es la de RWTH-BOSTON-104 (Zahedi et al., 2006), la cual está compuesta por videos en blanco y negro con una resolución de 312\*242 pixeles de 161 señas para el conjunto de entrenamiento y 40 señas para el conjunto de pruebas; las señas pertenecen a la lengua de señas americana y sus autores definen que los datos son para el reconocimiento de oraciones con señas continuas. Otra base de datos es la de SIGNUM (Agris y Kraiss, 2007), esta base de datos es una de las más ocupadas, corresponde a la lengua de señas alemana, consta de un vocabulario de 450 señas básicas con las cuales se conformaron 780 oraciones, las cuales fueron gesticuladas por 25 personas distintas; los videos suman un total de 56 horas y tienen una resolución de 776\*578 pixeles, los datos sirven para reconocimiento de palabras aisladas y de oraciones con señas continuas. ASLLVD (Athitsos et al., 2008) es una base de datos que consiste en más de 3000 videos generadas por 6 personas distintas con una resolución de 640\*480 pixeles, está base de datos contiene datos relacionados al deletreo de palabras, palabras aisladas y de oraciones compuestas que pueden ser ocupadas para reconocimiento de oraciones con señas continuas. La base que sirve como referencia en el área del reconocimiento de oraciones con señas continuas es la de RWTH-PHOENIX-Weather (Koller et al., 2015), la cual tiene videos extraídos de un noticiero donde se da información referente al clima mediante la lengua de señas alemana, la resolución de estos videos es de 210\*260 pixeles y se logró la extracción de 386 videos; esta base de datos es muy ocupada por el hecho de tratarse de una situación más realista, además de que porque viene anotada de forma muy completa.

Salvo en el caso de la base RWTH-PHOENIX-Weather donde se busco tener un entorno más realista, la mayoría de las bases de datos siguen protocolos de adquisición específicos en entornos controlados y con un numero limitado de personas. Con ello queda claro que los datos con los cuales se cuenta no son suficientes ni realistas, es por ello que investigaciones como las investigaciones de (Shi, Del Rio, Keane, Brentari, et al., 2019; Shi, Del Rio, Keane, Michaux, et al., 2019) han abordado enfoques llamados *in the wild*, obteniendo los videos que se han encontrado en servicios de redes sociales como YouTube, los cuales se presentan en situaciones mas realistas y con un número mayor de personas.

## 5 Discusión

En las secciones previas se habló de diversos tópicos que son necesarios para poder comprender que tipos de sistemas de reconocimiento automático existen, las características que los componen y las principales bases de datos que han sido ocupadas como referencia en el problema.

En el apartado de los sistemas que existen se presentaron tres tipos, los enfocados a reconocer deletreo de palabras, palabras aisladas y oraciones con señas continuas. El deletreo de palabras es un problema básico que no suele estudiarse tanto en la actualidad; la mayoría de la investigación se ha hecho sobre el reconocimiento de palabras aisladas; en años recientes se esta incrementado el interés hacia el reconocimiento de oraciones con señas continuas, pues este tipo de reconocimiento es el que será más útil para las aplicaciones que se buscan crear para ayudar en la comunicación de forma mas fluida entre la comunidad de sordos y la comunidad hablante, sin embargo la investigación en esta área esta en sus primeras fases, por lo que un área de oportunidad es hacer más trabajos relacionados con esta área.

Hablando de las características que componen a los sistemas de reconocimiento automático, en el punto referente a la adquisición de las imágenes se presentaron dos enfoques, uno basado en dispositivos de visión computacional y otro basado en sensores; si bien los que están basados en sensores tienen información más precisa, tienen el gran inconveniente de que algunos pueden ser incómodos, además, no cualquier usuario potencial puede tener acceso a ellos y principalmente se tiene la hipótesis de que su uso no es realista en el día a día, por lo cual debe de considerarse seriamente que tan útil es emplearlos. Por estas razones los enfoques que hacen uso de dispositivos de visión computacional son los más empleados, sin embargo, los trabajos que hacen uso del dispositivo Kinect también tienen que ser evaluados, pues independientemente de que provean información útil, también tienen el problema de que no son dispositivos de uso diario y que funcionen bien bajo en cualquier entorno (problemas de iluminación para la captura y necesidad de estar colocados de cierta forma), así los dispositivos de visión que capturan videos en 2D con formato RGB generalmente son los más usados; no obstante el uso de estos dispositivos de captura trae problemas que deben de tenerse en cuenta como lo son la resolución de los videos, la iluminación así como oclusiones, por lo cual un área de oportunidad es el que trabajo en esta área es ocupar métodos como los de super pixel para mejorar la resolución de videos, la eliminación del mayor ruido y de métodos adaptivos a través de técnicas de procesamiento de videos digitales.

En cuanto a las características que se suelen extraer, la mayoría de los trabajos aun se enfocan en las referentes a las que se extraen de las manos, esto porque se considera que es la zona que provee información mas relevante para el problema, pero en años recientes se esta explorando cada vez más la extracción de características no manuales, esto presenta varias áreas de oportunidad pues por si solo hay trabajo que aun se puede realizar, por ejemplo, en la estimación de la pose del cuerpo se pueden explorar métodos de aprendizaje profundo, en las expresiones fáciles se pueden generar nuevos modelos para definir las expresiones mas relevantes, pero independientemente de esos temas, la unión de las distintas características presenta una oportunidad para generar nuevos modelos de fusión de datos multimodales, pues hay que definir cual es la mejor forma de mezclar las características. Por otro lado, en el apartado de la etapa de clasificación hay un incremento en los métodos de aprendizaje profundo además de los métodos tradicionales, sin embargo, aún no se ha estudiado el uso de redes causales a profundidad y dada la

característica de tener un vocabulario controlado, el problema podría ser apto para el uso de estas redes, por lo cual podría ser un nuevo camino para explorar.

Finalmente, en el apartado de las bases de datos, si bien existen varias que pueden ser ocupadas para realizar pruebas y comparaciones con los trabajos de los distintos autores, aún no existe una base que contenga datos en el orden de los millones, en situaciones más realistas y en todos los lenguajes de señas, por lo cual la creación de más bases de datos con que contengan información en situaciones en contextos más comunes y no en ambientes tan controlados.

## 6 Conclusiones

En este trabajo se presentó una breve revisión para el problema del reconocimiento automático de lengua de señas, con ello se pretende presentar un punto inicial para futuras investigaciones. En particular se identificó que hay una necesidad de hacer más investigaciones en reconocimiento de oraciones con señas continuas, de incluir características multimodales, es decir manuales y no manuales, así como de presentar modelos sobre la mezcla de ellas, en particular un componente que no ha sido estudiado en profundidad es la de los movimientos de epéntesis. La correcta solución de todos estos retos representara un avance en esta área y por ende mejores aplicaciones y productos que ayuden a una mejor integración con la comunidad sorda.

## Referencias

- Agris, U. Von, y Kraiss, K.-F. (2007). "Towards a Video Corpus for Signer-Independent Continuous Sign Language Recognition". *GW 2007 The 7th International Workshop on Gesture in Human-Computer Interaction and Simulation*, pp. 10-11.
- Athitsos, V., Neidle, C., Sclaroff, S., Nash, J., Stefan, A., Yuan, Q., y Thangali, A. (2008). "The American Sign Language Lexicon Video Dataset". *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, June*.
- Bragg, D., Koller, O., Bellard, M., Berke, L., Boudrealt, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., Vogler, C., y Morris, M. R. (2019). "Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective". *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility*.
- Camgoz, N. C., Hadfield, S., Koller, O., y Bowden, R. (2017). "SubUNets: End-to-End Hand Shape and Continuous Sign Language Recognition". *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3075-3084.
- Cui, R., Liu, H., y Zhang, C. (2017). "Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization". *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1610-1618.
- Elakkiya, R., y Selvamani, K. (2019). "Subunit sign modeling framework for continuous sign language recognition". *Computers and Electrical Engineering*, vol. 74, pp. 379-390.



- Ferreira, P. M., Cardoso, J. S., y Rebelo, A. (2019). "On the role of multimodal learning in the recognition of sign language". *Multimedia Tools and Applications*, vol. 78, no. 8, pp. 10035-10056.
- Hu, Y. (2018). "Finger spelling recognition using depth information and support vector machine". *Multimedia Tools and Applications*, vol. 77, no. 21, pp. 29043-29057.
- Ibrahim, N. B., Zayed, H. H., y Selim, M. M. (2019). "Advances, Challenges and Opportunities in Continuous Sign Language Recognition". *Journal of Engineering and Applied Sciences*, vol. 15, no. 5, pp. 1205-1227.
- Jimenez, J., Martin, A., Uc, V., y Espinosa, A. (2017). "Mexican Sign Language Alphanumerical Gestures Recognition using 3D Haar-like Features". *IEEE Latin America Transactions*, vol. 15, no. 10, pp. 2000-2005.
- Kamal, S. M., Chen, Y., Li, S., Shi, X., y Zheng, J. (2019). "Technical Approaches to Chinese Sign Language Processing: A Review". *IEEE Access*, vol. 7, pp. 96926-96935.
- Koller, O., Forster, J., y Ney, H. (2015). "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers". *Computer Vision and Image Understanding*, vol. 141, pp. 108-125.
- Koller, O., Zargaran, S., Ney, H., y Bowden, R. (2018). "Deep Sign: Enabling Robust Statistical Continuous Sign Language Recognition via Hybrid CNN-HMMs". *International Journal of Computer Vision*, vol. 126, no. 12, pp. 1311-1325.
- Lim, K. M., Tan, A. W. C., Lee, C. P., y Tan, S. C. (2019). "Isolated sign language recognition using Convolutional Neural Network hand modelling and Hand Energy Image". *Multimedia Tools and Applications*, vol. 78, no. 14, pp. 19917-19944.
- Lim, K. M., Tan, A. W. C., y Tan, S. C. (2016). "Block-based histogram of optical flow for isolated sign language recognition". *Journal of Visual Communication and Image Representation*, vol. 40 (Part B), pp. 538-545.
- Mohandes, M., Deriche, M., y Liu, J. (2014). "Image-Based and Sensor-Based Approaches to Arabic Sign Language Recognition". *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 4, pp. 551-557.
- Shi, B., Del Rio, A. M., Keane, J., Brentari, D., Shakhnarovich, G., y Livescu, K. (2019). *Fingerspelling recognition in the wild with iterative visual attention*.
- Shi, B., Del Rio, A. M., Keane, J., Michaux, J., Brentari, D., Shakhnarovich, G., y Livescu, K. (2019). "American Sign Language Fingerspelling Recognition in the Wild". *2018 IEEE Spoken Language Technology Workshop, SLT 2018 - Proceedings*, pp. 145-152.
- Tanwar, V. K., Buckchash, H., Raman, B., y Bhargava, R. (2019). "Dense motion analysis of German finger spellings". *Multimedia Tools and Applications*, vol. 78, no. 8, pp. 9511-9536.
- Wadhawan, A., y Kumar, P. (2019). "Sign Language Recognition Systems: A Decade Systematic Literature Review". *Archives of Computational Methods in Engineering*, 0123456789.
- WDF, World Federation of the Deaf. (2020). *Our Work*. Recuperado de <https://wfdeaf.org/our-work/>.
- WHO, World Health Organization. (2020). *Deafness and hearing loss*. Recuperado de <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss>.
- Zahedi, M., Dreuw, P., Rybach, D., Deselaers, T., y Ney, H. (2006). "Geometric features for improving continuous appearance-based sign language recognition". *BMVC 2006 - Proceedings of the British Machine Vision Conference 2006*, pp. 1019-1028.
- Zieren, J., Canzler, U., Bauer, B., y Kraiss, K.-F. (2006). "Sign Language Recognition". In *Advanced Man-Machine Interaction*, pp. 95-139.

# Capítulo 18

## Minería y análisis de sentimientos de textos históricos: una revisión de la literatura

Stephanie Vázquez-González, María Somodevilla-García

Benemérita Universidad Autónoma de Puebla,  
Facultad de Ciencias de la Computación  
14 sur y Av. San Claudio, C.U., Puebla, Puebla, México  
stephanie.vazquez@gmail.com

**Resumen.** El análisis de textos de tipo histórico para identificar características específicas, así como para realizar un análisis de sentimientos es un área de estudio que converge dentro de las llamadas humanidades digitales. A continuación, se presenta una revisión de la principal literatura análisis automático de textos, análisis de sentimientos y algunos trabajos donde convergen estas áreas del procesamiento del lenguaje natural (PLN) con al análisis de textos históricos.

**Palabras clave:** Análisis de sentimientos, Minería de textos, Humanidades digitales, Textos históricos.

### 1 Introducción

Una de las principales aplicaciones prácticas del procesamiento del lenguaje natural (PLN) es la del análisis de textos. Para cualquiera de los objetivos que se plantee en PLN, es necesario que las computadoras sepan interpretar el contenido de un texto, por lo que se han desarrollado multitud de técnicas para analizar y extraer automáticamente el contenido de los mismos. Las llamadas humanidades digitales es un área donde convergen estas dos áreas de investigación, el análisis de textos históricos y el procesamiento automático de textos, es un término que engloba este campo interdisciplinario que busca entender el impacto y la relación de las tecnologías de cómputo en el quehacer de los investigadores en las humanidades.

El presente trabajo se organiza como sigue. En la sección 2, revisamos los conceptos teóricos del análisis automático de texto y del análisis de sentimientos. En la Sección 3, revisamos la aplicación del análisis automático de textos aplicado a textos históricos. En la Sección 4, explicamos cada etapa del estudio comparativo realizado, iniciando por la metodología diseñada y a continuación el desarrollo y datos encontrados en cada una de estas etapas. En la Sección 5, comentamos las conclusiones tras finalizar la revisión de los trabajos encontrados.

## 2 Análisis automático de textos y análisis de sentimientos

El análisis de texto, también conocido como minería de texto, es el proceso automatizado que utiliza inteligencia artificial (IA) para obtener información del texto. Algunas de las técnicas populares de análisis de texto incluyen análisis de sentimientos, detección de temas y extracción de palabras clave (Liu, 2012).

### 2.1 Análisis automático de textos

El análisis de texto combina nociones de estadística, lingüística y aprendizaje automático para crear modelos que aprenden de los datos de entrenamiento y pueden predecir resultados sobre nueva información en función de su experiencia previa.

Existen métodos básicos para el análisis de texto y otros más avanzados. Algunas de las técnicas más simples son la frecuencia de palabras, colocaciones y concordancia. Los métodos más avanzados se dividen principalmente en clasificación de texto y extracción de texto. (Cambria, Das, Bandyopadhyay, & Feraco, 2017)

La clasificación de texto tiene dos enfoques principales en la actualidad, estos son los sistemas basados en reglas y los sistemas basados en aprendizaje automático. Los sistemas basados en reglas detectan estos patrones lingüísticos anotados manualmente en los textos y asignan las etiquetas correspondientes en función de los resultados de las detecciones. Los sistemas basados en *Machine Learning* o aprendizaje automático pueden hacer predicciones basadas en lo que aprenden de observaciones pasadas.

Hay muchos algoritmos de aprendizaje automático utilizados en la clasificación de texto. Los más utilizados son la familia de algoritmos Naive Bayes (NB), *Support Vector Machines* (SVM) y algoritmos de aprendizaje profundo (DL). También existen algunos sistemas híbridos que generalmente contienen sistemas basados en aprendizaje automático en sus núcleos y sistemas basados en reglas que se utilizan para mejorar aún más las predicciones (Appel, Chiclana, Carter, & Fujita, 2016).

### 2.2 Análisis de sentimientos

El análisis de sentimientos, es un proceso automatizado de analizar datos de texto y clasificar opiniones como negativas, positivas o neutrales. Existen muchos tipos de análisis de sentimientos y las herramientas van desde sistemas que se centran en la polaridad (positiva, negativa, neutral) hasta sistemas que detectan emociones (enojado, feliz, triste, etc.) o identifican intenciones (por ejemplo, no interesado) (Pang & Lee, 2008).

Entre los principales tipos de análisis de sentimientos se pueden encontrar: el análisis de sentimiento fino que obtiene una mayor precisión en el nivel de polaridad; la detección de emociones es otro tipo y tiene como objetivo detectar emociones como felicidad, ira, tristeza y etc. Los métodos y algoritmos principales para análisis de sentimientos son los

mismos que para la clasificación general de textos: sistemas basados en reglas, sistemas automáticos y sistemas híbridos.

### **3 Análisis automático de textos históricos**

En cuanto al tema del análisis de textos históricos, se han hecho trabajos variados, desde la base misma para poder analizar textos históricos que es la digitalización de manuscritos para poder realizar búsquedas y procesamiento automático posteriormente ya que el desarrollo de investigación en el área de las humanidades digitales depende ampliamente de la disponibilidad de textos en formato digital (Piotrowski, 2012).

Otro tema que debe tomarse en consideración al trabajar con textos históricos es la traducción a la versión contemporánea de un idioma, para traducir la ortografía histórica a una ortografía moderna antes de aplicar las herramientas de PNL.

### **4 Estudio comparativo**

Para la creación del presente estudio comparativo se definió un protocolo con el conjunto de pasos a seguir para una revisión sistemática efectiva. Primero debemos definir el dominio de la literatura que se revisará, en este caso, artículos sobre minería de texto o análisis automático de texto y su caso particular que es el análisis de sentimientos, todo esto aplicado a textos largos a diferencia de la mayoría de los trabajos actuales que se centran. Y aún más particular decidimos buscar si existían trabajos de análisis de texto realizados a textos de carácter histórico.

#### **4.1 Metodología de revisión de la literatura**

A continuación, se enumeran los pasos que se definieron para la metodología de revisión de la literatura:

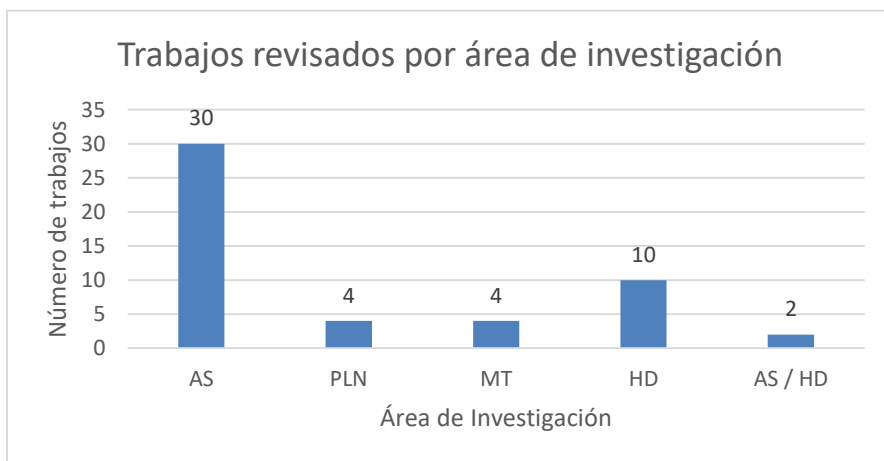
1. Definición del problema a investigar, los temas a considerar son las técnicas de análisis automático de textos y de análisis de sentimientos, y estas dos técnicas aplicadas a textos históricos.
2. Haciendo uso de preguntas de investigación surgidas del tema de investigación en específico, se puede definir mejor el propósito y finalidad de una revisión del estado del arte.
3. Detallar los pasos para el protocolo de búsqueda en la literatura encontrada con el fin de identificar las publicaciones principales y se definen los términos claves para la búsqueda de trabajos publicados. La estrategia principal, es incluir artículos científicos disponibles en plataformas digitales tales como Elsevier, ResearchGate, Scopus, Google Scholar, Academia, etc.

4. Hacer una selección de textos útiles para la investigación, detallando los criterios a seguir para incluir los trabajos encontrados.
5. Utilizar criterios de exclusión de trabajos.
6. Evaluar la calidad de los trabajos revisados para su inclusión.
7. Para la extracción de datos significativos de cada trabajo revisado se propone el uso de tablas comparativas que muestren los elementos que coinciden en los diferentes trabajos revisados.

#### 4.2 Trabajos revisados y sus áreas de investigación

Se hizo una revisión de alrededor de 50 trabajos relacionados con los temas de interés mencionados en el punto 1 de la sección anterior, los cuales se pueden clasificar principalmente en 4 áreas de investigación: Minería de Texto (MT), Procesamiento de Lenguaje Natural (PLN), Análisis de Sentimientos (AS) y Humanidades Digitales (HD).

Como se puede observar en la Figura 1, hay una cantidad significativamente más grande de trabajos relacionados con el área de AS que del resto de las áreas de interés.



**Fig. 1.** Número de trabajos encontrados referentes a cada área de investigación

Un listado de algunos de los trabajos revisados, así como algunos datos de cada uno están listados en la Tabla 1. Los trabajos revisados fueron obtenidos a través de las plataformas ya mencionadas en el punto número 3 y bajo los criterios para ser incluidos mencionados en los puntos 4, 5 y 6 de la metodología para revisión de la literatura de la sección anterior.

En todos los trabajos revisados hay diversas estrategias o enfoques para abordar su propio problema de investigación, pero podemos englobarlos principalmente en las siguientes categorías:

- Estudios comparativos entre diversas estrategias (para análisis de sentimientos, análisis de textos históricos);
- Estrategias basadas en lexicón para análisis de sentimientos;
- Estrategias que utilizan algoritmos de *machine learning* o *deep learning* para análisis de sentimientos o minería de texto;
- Revisión de algún sistema desarrollado para análisis de sentimientos (Lingmotif, SentiWordNet, SentProp);
- Diferentes niveles de granularidad en el análisis de sentimientos;
- Uso de herramientas de *machine learning* o *deep learning* para transcripción de manuscritos.

Podemos observar que no hay trabajos relacionados al análisis de sentimientos en textos históricos y la cantidad de trabajos encontrados respecto al análisis de sentimientos en textos extensos es limitada.

### 4.3 Objetivo de la investigación

En cuanto a los objetivos de la investigación clasificamos en 4 ramas principales los objetivos de los trabajos revisados, listadas a continuación:

- Comparativa de diferentes técnicas con un mismo objetivo;
- Medición del desempeño de una técnica en particular;
- Propuesta de una nueva técnica o modelo;
- Revisión de un tema de investigación

La comparativa de diferentes técnicas, por lo general utiliza alguna métrica estándar para medir el desempeño de cada una en la misma tarea. En la medición del desempeño de una sola técnica, también se utilizan por lo general métricas estándar para dar a conocer el desempeño comparada con investigaciones similares. En el caso de la propuesta de una nueva técnica también se utilizan métricas para comparar su desempeño con respecto a otras técnicas del estado del arte. Cuando se realiza revisión de un tema no necesariamente se mide el desempeño de las técnicas revisadas y por ello muchos de los trabajos de revisión no utilizan estas métricas mencionadas. En la sección 4.6 revisaremos las técnicas mencionadas utilizadas para medir el desempeño en cada una de las investigaciones encontradas para revisión. En la Tabla 2 podemos ver los objetivos de investigación de algunos de los trabajos analizados en la presente revisión de literatura.

#### 4.4 Datasets

Para el correcto funcionamiento de las estrategias presentadas en los trabajos a revisar es necesario entrenar el modelo o algoritmo con muchos datos. Para este propósito, los investigadores reúnen una cantidad considerable de texto en un corpus o *dataset*.

Muchos de los trabajos revisados en el presente trabajo utilizan conjuntos de datos o *datasets* para probar las técnicas presentadas y poder medir su eficiencia, algunos de estos *datasets* son públicos, pero también algunos *datasets* fueron construidos especialmente para poder probar la nueva técnica o algoritmo propuesto. Y hay otro grupo en particular de conjuntos de datos que ya existían previos a la realización de la investigación, pero no habían sido utilizados anteriormente para su análisis automático o en su caso su transcripción automática como es el caso de los *datasets* formados por documentos históricos. En la Tabla 3 podemos ver un listado y breve descripción de algunos de los datasets utilizados en los trabajos analizados.

#### 4.5 Metodología

En los trabajos de investigación relacionados con análisis automático de textos, que involucran el uso de técnicas de Procesamiento de Lenguaje Natural, se pueden identificar varios pasos que están presentes en la mayoría de estos proyectos (Alake, 2020). Para hacer una revisión de las metodologías seguidas, decidimos definir los siguientes pasos o etapas:

- **Definición del problema**, es la etapa inicial y se enfoca en adquirir una comprensión del problema a resolver aplicando algoritmos de inteligencia artificial. La comprensión del problema, las soluciones propuestas y los datos disponibles, permitirá un proceso de selección de modelo aprendizaje automático adecuado.
- **Recolección de datos**, es un paso crucial que sienta un precedente para la efectividad y el rendimiento de un modelo entrenado.
- **Pre-procesamiento de datos**, los pasos de pre-procesamiento de datos se basan principalmente en los requisitos de entrada del modelo.
- **Implementación del modelo**, por lo general, la implementación del modelo se simplifica al compararlo con los modelos existentes que están disponibles en una variedad de fuentes en línea.
- **Entrenamiento**, los datos obtenidos en las etapas anteriores se utilizan dentro de la etapa de entrenamiento.
- **Evaluación**, en esta etapa, se debe contar ya con un modelo entrenado y listo para realizar la evaluación de su desempeño.
- **Refinado de parámetros**, el ajuste de parámetros es el proceso de refinamiento del modelo que se realiza haciendo modificaciones a los valores de parámetros definidos.

- **Liberación del modelo**, esta es la etapa final del modelo dentro de todas las etapas identificadas. Integrar un modelo dentro de un ecosistema más amplio de aplicación o herramienta es un paso esencial de la implementación del modelo.

**Tabla 1.** Listado de los trabajos encontrados para esta revisión

No.	Título	Autores y año de publicación
1	Sentiment Analysis: A Comparative Study on Different Approaches	M. Devika, C. Sunitha, A. Ganesh (2016) (Devika, Sunitha, & Ganesh, 2016)
2	Generate domain-specific sentiment lexicon for review sentiment analysis	H. Han, J. Zhang, J. Yang et al. (2018) (Han, Zhang, Yang, Shen, & Zhang, 2018)
3	Deep learning for sentiment analysis: A survey	L. Zhang, S. Wang, B. Liu (2018) (Zhang, Wang, & Liu, 2018)
4	Lexicon based feature extraction for emotion text classification	A. Bandhakavi, N. Wiratunga, D. Padmanabhan et al. (2017) (Bandhakavi, Wiratunga, Padmanabhan, & Massie, 2017)
5	A sentiment classification model based on multiple classifiers	C. Catal, M. Nangir (2017) (Catal & Nangir, 2017)
6	Classification of sentiment reviews using n-gram machine learning approach	A. Tripathy, A. Agrawal, S. Rath (2016) (Tripathy, Agrawal, & Rath, 2016)
7	Lexicon-Based Methods for Sentiment Analysis	M. Taboada, J. Brooke, Milan Tofiloski et al. (2011) (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011)
8	Affective Computing and Sentiment Analysis	E. Cambria (2016) (Cambria, 2016)
9	Classification of Sentimental Reviews Using Machine Learning Techniques	A. Tripathy, A. Agrawal, S. Rath (2015) (Tripathy, Agrawal, & Rath, 2015)
10	Sentence-Level Emotion Detection Framework Using Rule-Based Classification	M. Asghar, A. Khan, A. Bibi et al. (2017) (Asghar, Khan, Bibi, Kundi, & Ahmad, 2017)

La Figura 2 muestra un gráfico de cuántos de los trabajos analizados cumplen con cada una de las etapas identificadas de la metodología, al ser trabajos de investigación científica todos cumplen al menos con las dos primeras etapas que son definición del problema e investigación, en cuanto a la recolección de datos solamente menos de la mitad de los trabajos de investigación (24) se ocupó de recabar sus datos propios, y algunos más utilizaron conjuntos de datos públicos recabados durante trabajos de investigación previos.



**Tabla 2.** Objetivo identificado de cada trabajo en la revisión

Objetivo de la investigación					
No.	Título	Comparativa de dif. técnicas	Medición de desempeño una técnica	Propuesta nueva técnica /modelo	Revisión del tema
1	Sentiment Analysis: A Comparative Study on Different Approaches	x			X
2	Generate domain-specific sentiment lexicon for review sentiment analysis		x		
3	Deep learning for sentiment analysis: A survey	x			X
4	Lexicon based feature extraction for emotion text classification		x		
5	A sentiment classification model based on multiple classifiers			x	
6	Classification of sentiment reviews using n-gram machine learning approach	x			
7	Lexicon-Based Methods for Sentiment Analysis		x	x	
8	Affective Computing and Sentiment Analysis				X
9	Classification of Sentimental Reviews Using Machine Learning Techniques	x	x		
10	Sentence-Level Emotion Detection Framework Using Rule-Based Classification		x	x	

**Tabla 3.** Listado de los conjuntos de datos utilizados en los trabajos revisados

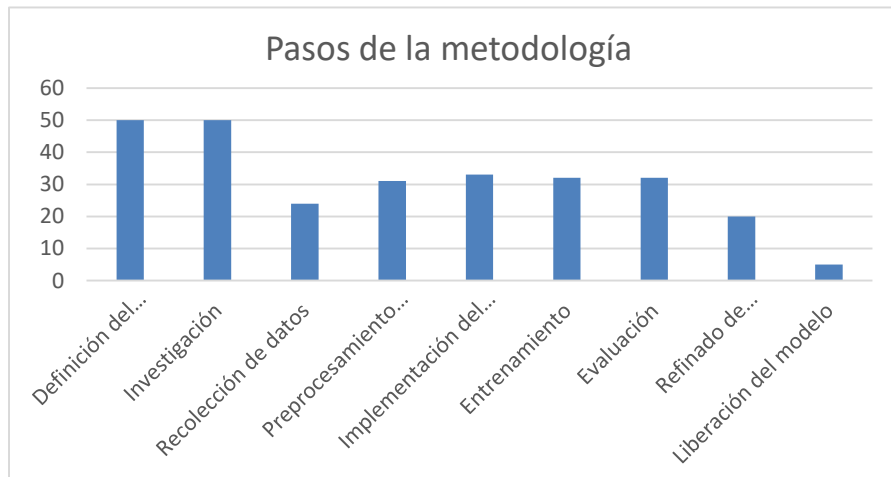
Conjunto de datos utilizado (Dataset)	Descripción
Large Movie Review Dataset y Amazon Product Review Dataset (Han et al., 2018)	Reseñas de películas y productos con 50,000 y 8,000 entradas respectivamente
Benchmark datasets (Bandhakavi et al., 2017)	1250 noticias emocionales, 280,000 tuits emocionales, 5500 entradas de blogs y 7000 reportes de incidentes
Conjunto de datos en idioma turco (Catal & Nangir, 2017)	Reseñas de libros, películas y productos
IMDB dataset (Tripathy et al., 2016)	Reseñas de películas con 100,000 entradas para entrenamiento y prueba
Epinions 1 y 2, Polarity Dataset, Camera Dataset (Taboada et al., 2011)	400 reseñas de dif productos, 1900 textos con polaridad de películas y 2400 reseñas de prods, respectivamente
Polarity Movie Dataset (Tripathy et al., 2015)	1000 reseñas positivas y 1000 reseñas negativas de películas
Tweets, ISEAR, Pang and Lee (Asghar et al., 2017)	Más de 10,000 tuits o sentencias sobre noticias y productos y datasets gold standard respectivamente
RepLab 2010 / Tweets (De Albornoz, Chugur, & Amigó, 2012)	18000 tuits sobre 6 compañías
Conjunto de datos de canciones (Moreno Arboleda, Duitama Muñoz, & Montoya Gómez, 2016)	540 canciones de 18 categorías diferentes
Alcide De Gasperi Dataset (Sprugnoli, Tonelli, Marchetti, & Moretti, 2015)	Escritos de Alcide De Gasperi, consta de 3000 textos y 3,000,000 de palabras publicado entre 1901 y 1954

La mayoría de los trabajos que implementaron un modelo (33) también realizaron un pre-procesamiento de los datos para mejores resultados (31), y en general los trabajos que no realizaron la implementación de un modelo se debe a que son trabajos de revisión únicamente. Finalmente, la liberación de un modelo para uso público ya sea en plataformas web o como software para su distribución solo ocurrió en un número muy reducido de casos (5).

#### 4.6 Resultados y métricas

A partir de la etapa de evaluación de los proyectos revisados ya se pueden obtener datos de los resultados producidos por un modelo, para medir la eficiencia de dichos resultados se utilizan medidas estándar, entre las más comunes tenemos las que se mencionan a continuación: accuracy, precision, recall, f1-score. En la Figura 3 podemos ver la variedad de medidas utilizadas, es notable que el uso de accuracy es la métrica más común, aunque no sea la mejor medida para rendimiento. En general, es más efectivo utilizar F1 score, que

es la que le sigue en popularidad en los trabajos revisados, y a continuación las medidas de precisión y recall.



**Fig. 2.** Gráfico que muestra cuantos de los trabajos revisados cumplen con cada etapa de la metodología identificada

## 5 Conclusiones

Tras revisar los trabajos relacionados a las áreas de investigación de interés se puede notar que, aunque todos tienen el rigor científico requerido, y cumplen con las etapas de definición del problema e investigación, muy pocos llegan hasta la etapa de liberación del modelo (5) y sólo algunos están al alcance de usuarios interesados en ese tipo de herramientas. También algo que llama la atención son los *datasets* utilizados, en varios casos se emplearon *datasets* públicos hechos para entrenamiento, y en el casi la mitad de los casos se construyeron *datasets* con características particulares como en el caso del análisis de textos históricos o manuscritos (24). En cuanto a los objetivos la mayor parte de los trabajos (32) se centran en presentar una nueva técnica o modelo para resolver un problema específico. Y como conclusión final se puede notar la ausencia de trabajos de análisis de sentimiento en textos históricos, limitándose a muy pocos trabajos bien reconocidos. Por ello sería de suma importancia realizar una investigación donde convergen las áreas de humanidades, representadas por los textos históricos y su análisis automático de parte de las tecnologías y converger en un trabajo multidisciplinario que tiene cabida dentro del área de investigación de las humanidades digitales.



**Fig. 3.** Grafico que muestra las distintas métricas usadas para medir el desempeño de los trabajos revisados

## Referencias

- Alake, R. (2020). *10 Stages Of A Machine Learning Project In 2020 (And Where You Fit)*. 2020, 1–17. Retrieved from <https://towardsdatascience.com/10-stages-of-a-machine-learning-project-in-2020-and-where-you-fit-cb73ad4726cb>
- Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2016). A hybrid approach to the sentiment analysis problem at the sentence level. *Knowledge-Based Systems*. <https://doi.org/10.1016/j.knsys.2016.05.040>
- Asghar, M. Z., Khan, A., Bibi, A., Kundi, F. M., & Ahmad, H. (2017). Sentence-Level Emotion Detection Framework Using Rule-Based Classification. *Cognitive Computation*, 9(6), 868–894. <https://doi.org/10.1007/s12559-017-9503-3>
- Bandhakavi, A., Wiratunga, N., Padmanabhan, D., & Massie, S. (2017). Lexicon based feature extraction for emotion text classification. *Pattern Recognition Letters*. <https://doi.org/10.1016/j.patrec.2016.12.009>
- Cambria, E. (2016). Affective Computing and Sentiment Analysis. *IEEE Intelligent Systems*, 31(2). <https://doi.org/10.1109/MIS.2016.31>
- Cambria, E., Das, D., Bandyopadhyay, S., & Feraco, A. (2017). *A Practical Guide to Sentiment Analysis (Socio-Affecting Computing 5)*.
- Catal, C., & Nangir, M. (2017). A sentiment classification model based on multiple

- classifiers. *Applied Soft Computing Journal*.  
<https://doi.org/10.1016/j.asoc.2016.11.022>
- De Albornoz, J. C., Chugur, I., & Amigó, E. (2012). Using an emotion-based model and sentiment analysis techniques to classify polarity for reputation. *CEUR Workshop Proceedings*, 1178. Retrieved from <http://www.reputationinstitute.com/thought-leadership/>
- Devika, M. D., Sunitha, C., & Ganesh, A. (2016). Sentiment Analysis: A Comparative Study on Different Approaches. *Procedia Computer Science*, 87. <https://doi.org/10.1016/j.procs.2016.05.124>
- Han, H., Zhang, J., Yang, J., Shen, Y., & Zhang, Y. (2018). Generate domain-specific sentiment lexicon for review sentiment analysis. *Multimedia Tools and Applications*, 77(16), 21265–21280. <https://doi.org/10.1007/s11042-017-5529-5>
- Liu, B. (2012). Sentiment analysis and opinion mining. In *Synthesis Lectures on Human Language Technologies* (Vol. 5). <https://doi.org/10.2200/S00416ED1V01Y201204HLT016>
- Moreno Arboleda, F. J., Duitama Muñoz, J. F., & Montoya Gómez, L. F. (2016). Una propuesta para la clasificación emocional de un álbum a partir de la letra de sus canciones. *Revista Tecnura*, 20(47), 57. <https://doi.org/10.14483/udistrital.jour.tecnura.2016.1.a05>
- Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval*, 2(2), 1–135. <https://doi.org/10.2200/S00416ED1V01Y201204HLT016>
- Piotrowski, M. (2012). Natural language processing for historical texts. *Synthesis Lectures on Human Language Technologies*. <https://doi.org/10.2200/S00436ED1V01Y201207HLT017>
- Sprugnoli, R., Tonelli, S., Marchetti, A., & Moretti, G. (2015). Towards sentiment analysis for historical texts. *Digital Scholarship in the Humanities*, 31(4), 1–11. <https://doi.org/10.1093/lc/fqv027>
- Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-Based Methods for Sentiment Analysis. *Computational Linguistics*, 37(2), 267–307. Retrieved from [https://www.mitpressjournals.org/doi/pdfplus/10.1162/COLI\\_a\\_00049](https://www.mitpressjournals.org/doi/pdfplus/10.1162/COLI_a_00049)
- Tripathy, A., Agrawal, A., & Rath, S. K. (2015). Classification of Sentimental Reviews Using Machine Learning Techniques. *Procedia Computer Science*, 57, 821–829. <https://doi.org/10.1016/j.procs.2015.07.523>
- Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*. <https://doi.org/10.1016/j.eswa.2016.03.028>
- Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4). <https://doi.org/10.1002/widm.1253>

# Capítulo 19

## Evaluación comparativa de sistemas de reconocimiento automático del habla

Víctor Manuel Morales-de-Jesús, María Josefa Somodevilla-García

Facultad de Ciencias de la Computación  
Benemérita Universidad Autónoma de Puebla  
victor.morales@cs.buap.mx, mariajsomodevilla@gmail.com

**Resumen** El reconocimiento automático del habla se refiere al proceso de transcribir a texto las emisiones realizadas a través del habla. Esta tarea ha sido de gran interés desde hace varias décadas atrás. Pero es en tiempos recientes cuando ha surgido un desarrollo de sistemas de reconocimiento que realizan un desempeño aceptable al transcribir el habla a texto en distintos idiomas, entre ellos el español. Sin embargo, la precisión de estos sistemas varía considerablemente tomando en cuenta distintos aspectos como la calidad de audio de entrada, vocabulario empleado y diferencias en las características de voz de los hablantes. En este sentido, este trabajo realiza una evaluación comparativa de tres sistemas de reconocimiento automático del habla empleando un conjunto de datos de habla en español de México de personas de la tercera edad con el objetivo de medir la precisión mostrada en este contexto en particular. Como resultado de la evaluación se observa que la precisión del sistema de Google Speech-to-Text es más alta a los otros sistemas comparados obteniendo un valor WER de 0.268.

**Palabras Clave:** Sistemas ASR, reconocimiento automático del habla, WER

### 1. Introducción

El término de reconocimiento automático del habla o Automatic Speech Recognition (ASR) se refiere al proceso llevado a cabo de forma automática (por un dispositivo o computadora) de convertir la señal que emite la voz de una persona en su correspondiente secuencia de palabras y otras entidades lingüísticas, es decir, realizar una transcripción a texto (Li, Deng, Haeb-Umbach y Gong, 2016). Uno de los principales objetivos del ASR es desarrollar las herramientas que permitan una interacción humano-máquina más natural a través de la voz.

Desde hace ya varias décadas se ha mostrado un gran interés por poder llevar a cabo una interacción verbal con las máquinas (Shearman y Leach, 1968; White y Neely, 1976). Para que esto sea posible es necesario que sean capaces de poder escuchar y entender el lenguaje humano. La primera parte de este proceso es precisamente el que se refiere al ASR. Actualmente, se ha hecho un importante avance en este tema, por lo que existen sistemas ASR que permiten realizar

reconocimiento del habla con una elevada precisión. Este reconocimiento no conlleva la interpretación y comprensión del significado del texto transcrito, debido a que esta tarea se realiza posteriormente mediante técnicas de procesamiento de lenguaje natural.

En el presente trabajo se realiza una evaluación comparativa de distintos sistemas de ASR comerciales, tomando en cuenta aquellos que mencionan tener una precisión arriba del 85% de precisión en el reconocimiento del idioma español. Sin embargo, debido a las características del proyecto de doctorado planteado, es necesario analizar el desempeño de estos sistemas en reconocimiento de habla en personas mayores. En este sentido, la evaluación se lleva a cabo empleando archivos de audio que contienen conversaciones con personas de la tercera edad con el fin de observar el desempeño de estos sistemas ante situaciones que se pueden presentar en este grupo poblacional como pueden ser: 1) pausas largas al pronunciar la siguiente palabras u oración, 2) un mayor número de inicios falsos, 3) variaciones en la entonación, por mencionar algunos.

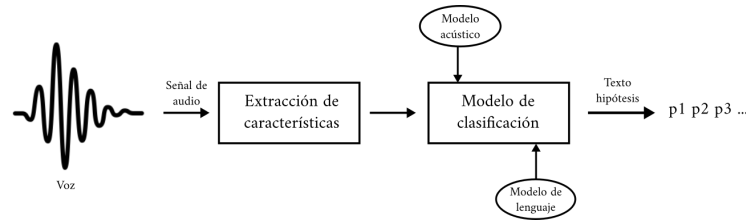
Este trabajo esta organizado de la siguiente manera: en la sección 2 se brinda una breve descripción de la tarea de reconocimiento automático del habla y se mencionan las características de los sistemas ASR comerciales que fueron analizados en este trabajo, posteriormente en la sección 3 se muestran los experimentos realizados y los resultados obtenidos. Finalmente, en la sección 4 se incluyen las conclusiones finales y el trabajo futuro.

## 2. Reconocimiento automático del habla

El reconocimiento automático del habla es un caso especial de reconocimiento de patrones. La figura 1 muestra las etapas de procesamiento que se llevan a cabo para realizar ASR. Existen dos fases en el reconocimiento de patrones supervisado, que son la fase de entrenamiento y la fase de prueba. Durante la fase de entrenamiento, los parámetros para crear el modelo de clasificación son estimados a partir de un gran número de instancias de las clases objetivo (datos de entrenamiento). Durante la fase de prueba o evaluación del modelo, las características de las instancias nuevas (datos de prueba) son analizadas empleando el modelo entrenado, comparando con todas y cada una de las clases candidatas. Finalmente, la instancia nueva es asignada a una clase a la cual el modelo de clasificación considera que se ajusta mejor.

El objetivo de realizar ASR es generar la secuencia de palabras óptima sujeta a restricciones lingüísticas. La oración se compone de unidades lingüísticas como palabras, sílabas, fonemas. La evidencia acústica proporcionada por los modelos acústicos de tales unidades se combina con las reglas de construcción de oraciones válidas y significativas en el lenguaje para crear una oración hipótesis. Por lo tanto, en el caso del reconocimiento de habla, la etapa de coincidencia de patrones puede verse en dos dominios: acústico y simbólico.

En el dominio acústico, un vector de características correspondiente a un pequeño segmento de prueba de voz se compara con el modelo acústico de todas y cada una de las clases. Al segmento se le asigna un conjunto de etiquetas de



**Fig. 1.** Diagrama del funcionamiento de un sistema ASR típico

las clases a las que tiene una mayor probabilidad de pertenencia. Este proceso de asignación de etiquetas se repite para cada vector de características de los datos de prueba. El conjunto resultante de hipótesis (oraciones candidatas) se procesa junto con el modelo de lenguaje para producir la oración reconocida.

## 2.1. Sistemas ASR

Los sistemas ASR hacen más fácil que una persona con alguna limitante física pueda controlar o interactuar con algún tipo de dispositivo. Incluso personas comunes preferirían interactuar con una computadora mediante en lugar de utilizar un teclado o un ratón. Debido al rápido desarrollo de sistemas ubicuos, la interfaz por voz resulta cada vez más conveniente como método de entrada de datos al sistema en distintas circunstancias. Algunos casos donde la aplicación de ASR resulta útil pueden ser: sistemas de dictado (Poder, Fisette y Déry, 2018), subtítulo automático (Che, Luo, Yang y Meinel, 2017), traducción automática (Nasereddin y Omari, 2017), búsqueda por voz y recientemente la interacción con asistentes inteligentes (Wolters, Kelly y Kilgour, 2016). A continuación se describen los sistemas ASR analizados dentro de este trabajo.

**Google Speech-to-Text (ASR-1)** Google Cloud Speech-to-Text <sup>1</sup> es un sistema que permite a los usuarios procesar archivos de audio cortos, extensos y audio en tiempo real que contenga habla y obtener como respuesta la transcripción escrita del habla contenida en el audio procesado. Actualmente, este sistema reconoce más de 120 idiomas y sus variantes, entre ellos el español de México. Sin embargo, hasta este momento existen características que solo pueden analizarse en el idioma inglés como: puntuación automática, especificación del contexto y reconocimiento del hablante.

**IBM Watson Speech to Text (ASR-2)** El sistema ASR que ofrece IBM Watson Speech to Text <sup>2</sup> permite a los usuarios seleccionar entre un modelo de banda ancha (audio con un muestreo mayor a 16 kHz.) y un modelo de banda

<sup>1</sup> <https://cloud.google.com/speech-to-text>

<sup>2</sup> <https://www.ibm.com/cloud/watson-speech-to-text>



estrecha (audio con muestreo a 8 kHz.) para procesar archivos de audio. Lo que permite obtener un buena relación consumo-rendimiento dependiendo del tipo de audio que se desea procesar. Este sistema permite la personalización del modelo acústico y el modelo de lenguaje a utilizar, así como limitar o extender la gramática del reconocedor a partir de un corpus brindado por el usuario de acuerdo al contexto de uso del sistema. Cuenta con soporte para 11 idiomas incluido el español en 5 variantes, entre ellas el español de México (aún en Beta).

**Amazon Transcribe (ASR-3)** Algunas de las características más relevantes del servicio ofrecido por Amazon Transcribe <sup>3</sup> es que permite que el usuario pueda crear y administrar diferentes vocabularios para poder ajustarlos de acuerdo al contexto de implementación. También brinda la posibilidad de asignar marcas temporales en cada palabra del texto, filtrar el vocabulario de la transcripción, identificación de canales en el audio y el reconocimiento de varios oradores. Este sistema permite la transcripción en 31 idiomas diferentes incluidos el español. Sin embargo, a diferencia de los dos sistemas anteriores no cuenta con un modelo para el español de México en específico.

Cada uno de los sistemas anteriores tiene distintas características que ofrecen una mayor usabilidad y un mejor desempeño en determinados escenarios. Sin embargo, como se mencionó anteriormente para este contexto en específico en donde se plantea implementar un sistema ASR dentro de un sistema de diálogo enfocado a personas con Alzheimer, que en su mayoría son personas de la tercera edad, es necesario evaluar el desempeño de los sistemas ASR dentro de este escenario. Por lo que a continuación se describen los experimentos realizados y los resultados obtenidos al evaluar estos sistemas.

### 3. Experimentos y resultados

En esta sección se muestra por una parte el proceso de la recolección del conjunto de datos y por otra parte los experimentos y resultados obtenidos.

#### 3.1. Conjunto de datos

Los archivos de audio para evaluar los sistemas ASR fueron recabados a partir de un cuestionario realizado a 8 personas consideradas adultos mayores, es decir, con una edad mayor a 65 años. Cada cuestionario fue grabado y su uso para esta investigación esta bajo el consentimiento de cada participante. Las preguntas planteadas en el cuestionario fueron las siguientes:

- ¿En qué año nació?
- ¿En dónde nació?
- Describa brevemente algo que le guste del lugar donde nació

---

<sup>3</sup> <https://aws.amazon.com/transcribe/>

- ¿Puede mencionar algún recuerdo agradable de su niñez?
- Describa brevemente su rutina en un día normal

Dentro del archivo de cada archivo de audio fueron eliminadas las preguntas, dejando únicamente las respuestas brindadas por cada participante. En la tabla 1 se muestran las características de los archivos de audio de acuerdo a cada participante.

**Tabla 1.** Descripción de archivos de audio

Archivo	Sexo	Edad	Duración (minutos)
AP_1	Femenino	67	3.28
AP_2	Masculino	68	5.32
AP_3	Femenino	65	4.15
AP_4	Femenino	81	4.10
AP_5	Masculino	78	6.18
AP_6	Femenino	76	3.14
AP_7	Femenino	67	5.50
AP_8	Masculino	70	4.30

Como se puede observar la mayoría de los participantes son mujeres ( $n = 5$ ) y la edad de los participantes se encuentra en un rango entre 65 y 81 años; la duración total de los archivos de audio recolectados es de 36.37 minutos. Es importante mencionar que todos los participantes en las entrevistas son personas que aun cuentan con sus habilidades comunicativas bien preservadas y sin ninguna enfermedad detectada que afecte su capacidad cognitiva y de lenguaje.

### 3.2. Experimentos

Para evaluar la precisión de la transcripción de un sistema ASR una métrica comúnmente usada es la tasa de error por palabra o Word Error Rate (WER) (Park, Patwardhan, Visweswariah y Gates, 2008; Ramírez Sánchez, Montalvo Bereau y Calvo de Lara, 2019), el valor WER es un valor numérico que se calcula mediante la formula 1:

$$WER = \frac{S + I + D}{N} \quad (1)$$

donde:

- $S$  representa el número de sustituciones, es decir, cada vez que una palabra es reemplazada (por ejemplo: “siempre queríamos salir” es transcrita como “siempre queríamos abrir”).
- $I$  representa el número de inserciones, es decir, cada vez que se agrega una palabra que no se dijo (por ejemplo: “comía temprano” es transcrita como “comía de temprano”).

- $D$  representa el número de eliminaciones, es decir, cada vez que una palabra es omitida o eliminada (por ejemplo: “caminaba con mi perro” es transcrita como “caminaba mi perro”).
- $N$  representa el número total de palabras que fueron transcritas.

Como se puede observar, por lo tanto el valor WER es el número de veces que el sistema ASR sustituyó, insertó o eliminó palabras del texto de referencia entre el número total de palabras transcritas. En este sentido, un valor de  $WER = 0$  indicaría una transcripción perfecta, mientras que un valor elevado indica un baja precisión por parte del sistema ASR.

Para evaluar los sistemas ASR inicialmente se realizó la transcripción manual de cada archivo de audio, las cuales son tomados como *transcripción de referencia* para comparar la transcripción brindada (transcripción hipótesis) por cada sistema. Dentro de la transcripción manual se conservaron palabras que pudieran ser casos de disfluencia que comúnmente se presentan al hablar considerando que al ser pronunciadas dentro del habla, el sistema ASR debería de ser capaz de reconocerlas.

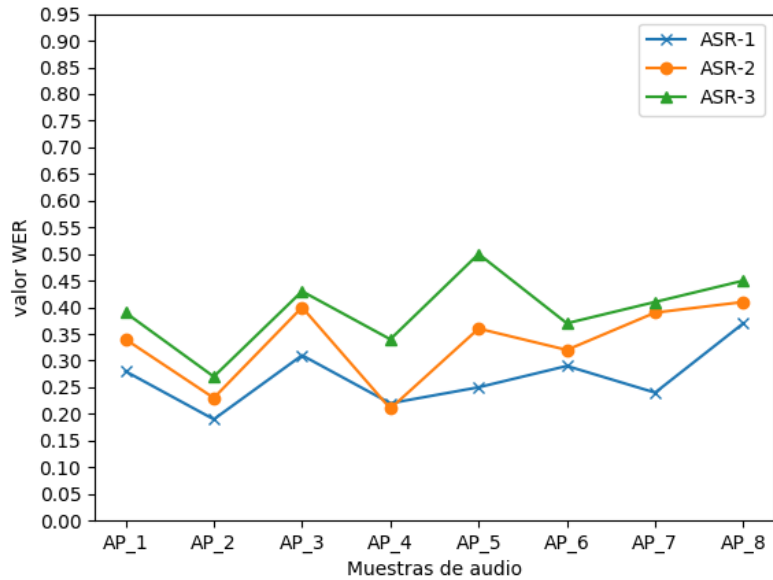
Posteriormente cada archivo de audio fue procesado por cada uno de los tres sistemas ASR y la transcripción hipótesis generada por cada sistema fue comparada con la transcripción de referencia y de esta forma se calculó el valor WER para cada archivo analizado.

**Tabla 2.** Valores WER obtenidos para cada archivo de audio

<b>Archivo</b>	<b>ASR-1</b>	<b>ASR-2</b>	<b>ASR-3</b>
AP_1	0.28	0.34	0.39
AP_2	0.19	0.23	0.27
AP_3	0.31	0.40	0.43
AP_4	0.22	0.21	0.34
AP_5	0.25	0.36	0.50
AP_6	0.29	0.32	0.37
AP_7	0.24	0.39	0.41
AP_8	0.37	0.41	0.45
Promedio	<b>0.268</b>	0.332	0.395

La tabla 2 muestra los valores WER para cada archivo, así como el promedio del valor WER para cada sistema. Se puede observar que el sistema de Google Speech-to-Text (ASR-1) muestra una mejor precisión al tener el menor valor promedio de WER (0.268). De igual forma, en la figura 2, se puede observar que el desempeño del sistema ASR-1 mostró un comportamiento más estable durante el procesamiento de todos los archivos de audio.

Los resultados obtenidos permiten establecer que el sistema de Google Speech-to-Text obtiene la mejor precisión en comparación con los otros dos sistemas.



**Fig. 2.** Comparación de los valores WER obtenidos

#### 4. Conclusiones y trabajo a futuro

Dentro de este trabajo se realizó una evaluación comparativa de los sistemas ASR comerciales que mencionan obtener una precisión por arriba del 85% en el idioma español. De los resultados obtenidos se observa que en este caso, donde la transcripción que se realizó fue de habla de personas de la tercera edad, la precisión mostrada por cada uno de los sistemas fue menor a la mencionada por cada uno de ellos. Sin embargo, el sistema que obtuvo un mejor desempeño mostró un comportamiento estable durante la transcripción de cada archivo. Por lo que se podría concluir que hasta este momento es el que brinda la mayor precisión en este caso en particular.

Como trabajo futuro, se plantea incorporar un mayor número de audios con una mayor diversidad en la entonación y cadencia del habla para observar el comportamiento de estos sistemas dentro de un conjunto más diverso. Además, se considera ampliar la comparación a sistemas ASR no comerciales que se enfocan en el idioma español.

## Referencias

- Che, X., Luo, S., Yang, H., y Meinel, C. (2017). “Automatic lecture subtitle generation and how it helps”. En *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*, (pp. 34–38). IEEE.
- Li, J., Deng, L., Haeb-Umbach, R., y Gong, Y. (2016). “Fundamentals of speech recognition”. En J. Li, L. Deng, R. Haeb-Umbach, y Y. Gong (Eds.), *Robust Automatic Speech Recognition* (pp. 9 – 40). Oxford: Academic Press.
- Nasereddin, H. H. y Omari, A. A. R. (2017). “Classification techniques for automatic speech recognition (ASR) algorithms used with real time speech translation”. En *2017 Computing Conference*, (pp. 200–207). IEEE.
- Park, Y., Patwardhan, S., Visweswariah, K., y Gates, S. C. (2008). “An empirical analysis of word error rate and keyword error rate”. En *Ninth Annual Conference of the International Speech Communication Association*, (pp. 2070–2073).
- Poder, T. G., Fiset, J.-F., y Déry, V. (2018). “Speech recognition for medical dictation: overview in Quebec and systematic review”. *Journal of medical systems*, vol. 42(5), pp. 89.
- Ramírez Sánchez, J. M., Montalvo Bereau, A. R., y Calvo de Lara, J. R. (2019). “Evaluación de Rasgos Acústicos para el Reconocimiento Automático del Habla en Escenarios Ruidosos usando Kaldi”. *Ingeniería Electrónica, Automática y Comunicaciones*, vol. 40, pp. 51 – 71.
- Shearman, J. y Leach, P. (1968). “Some experiments with a simple word recognition system”. *IEEE Transactions on Audio and Electroacoustics*, vol. 16(2), pp. 256–261.
- White, G. y Neely, R. (1976). “Speech recognition experiments with linear prediction, bandpass filtering, and dynamic programming”. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24(2), pp. 183–188.
- Wolters, M. K., Kelly, F., y Kilgour, J. (2016). “Designing a spoken dialogue interface to an intelligent cognitive assistant for people with dementia”. *Health informatics journal*, vol. 22(4), pp. 854–866.

# Capítulo 20

## Optimización para mejora de imágenes médicas

Eustolia Carreón-Esteban, J. Arturo Olvera-López, Ivan Olmos-Pineda

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación  
euce79@gmail.com, {aolvera,iolmos}@cs.buap.mx

**Resumen** La mejora de imágenes es una tarea perteneciente al área de procesamiento de imágenes digitales cuya finalidad es incrementar la calidad visual de las mismas, remarcando detalles que a simple vista no es posible distinguir. Es importante llevar a cabo este proceso para que las imágenes digitales sean usadas en sistemas de computadora que actualmente coadyuvan en el diagnóstico médico, haciéndolo más certero y confiable. Si bien es cierto que diversas alternativas para la solución de este problema han sido propuestas por muchos investigadores interesados en el área, siendo la ecualización del histograma la más usada y explorada, esta tiene algunas desventajas como el sobreajuste del brillo y del contraste, lo cual ocasiona que una imagen pierda naturalidad e incluso distorsión de bordes. En este trabajo se presenta una propuesta de mejora de imágenes médicas desde el enfoque de optimización mediante un algoritmo de lobos grises, el cual se encarga de realizar la búsqueda de parámetros que ajusten el contraste de la imagen para hacerla más adecuada en tareas posteriores de procesamiento digital.

**Palabras Clave:** Mejora de Imágenes · Optimizador de lobos grises · Imágenes Médicas.

### 1. Introducción

En la actualidad, el procesamiento de imágenes digitales es un área de investigación de gran interés debido a que las tareas que se incluyen tienen diversas aplicaciones, por ejemplo en la medicina. Aunque el avance tecnológico ha influido en la obtención de imágenes digitales de alta calidad, aún hay factores que repercuten, tales como la iluminación, el proceso de digitalización (Sonali et al, 2019)(Long et al, 2018) el campo magnético de los dispositivos, los movimientos en la toma de la muestra, la proyección del músculo, entre otros. Estos aspectos pueden generar como resultado niveles de gris imprecisos o bajo contraste, por lo tanto, en otras tareas de procesamiento resultan poco confiables los procesos de segmentación, extracción de características y clasificación, lo que además puede conducir a un diagnóstico médico inexacto (Chaira, 2014).

Por lo anterior, uno de los problemas a resolver en esta área es la mejora de imágenes, cuya finalidad es incrementar la calidad visual para ser usadas posteriormente en sistemas informáticos para su análisis, interpretación y obtención

de información. Este proceso lleva a humanos expertos en determinada área a hacer conclusiones y a tomar decisiones de forma más rápida y certera. Debido a que existen diferentes enfoques para la solución a este problema, no hay un método único para mejorar imágenes digitales. A través del tiempo se han hecho diversas propuestas que van desde métodos basados en el dominio espacial, siendo el más utilizado la ecualización del histograma; hasta aquellos que usan el dominio de la frecuencia como la transformada de Fourier y wavelets, entre otros.

En este trabajo se presenta el uso del algoritmo Optimizador de Lobos Grises para hallar los parámetros más adecuados de la función de transformación y evaluar su desempeño en la mejora de imágenes médicas.

## 2. Preliminares

Recientemente los algoritmos de optimización inspirados en la naturaleza se han empleado ampliamente en la mejora de la imágenes para encontrar los parámetros óptimos de diferentes funciones de transformación al minimizar o maximizar las funciones objetivo apropiadas. Estos algoritmos hacen uso de operadores aleatorios que actúan sobre un conjunto de soluciones candidatas, generan nuevas soluciones en un espacio de búsqueda y posteriormente son evaluadas con base en su aptitud (grado de aportación a la solución del problema). De acuerdo a los resultados se determina si se reemplaza la solución actual. Este proceso es repetido hasta cumplir con un criterio de terminación, el cual puede ser un número de generaciones, o hasta encontrar la mejor aproximación a la solución deseada. Los Algoritmos Genéticos (Munteanu y Rosa, 2004)(Hoseini y Shayesteh, 2013)(Ebenezer y Jude, 2015), la Evolución Diferencial (Dos Santos et al,2009)(Kumar et al, 2014, Particle Swarm Optimization (PSO) (Shanmugavadivu et al,2014)(Nickfarjam y Ebrahimpour-Komleh, 2017)(Zhiwei et al, 2015, así como Ant Colony Optimization (ACO) (Hoseini y Shayesteh, 2013) (Gao y Zeng, 2015), son los algoritmos que más se han explorado en el área de mejora de imágenes, dando buenos resultados en aquellas pertenecientes al mundo real. Por otra parte existen nuevas propuestas de algoritmos metaheurísticos cuya aplicación en el campo de las imágenes está empezando a explorarse; sin embargo, en otras áreas han tenido resultados competentes, e incluso mejores que otros algoritmos, tal es el caso del Optimizador de Lobos Grises (GWO-Grey Wolf Optimizer).

### 2.1. Mejora de imágenes

La mejora de imágenes en el ámbito espacial usa una función de transformación para generar nuevos valores de intensidad para cada uno de los pixeles dentro de la imagen. Desde un punto de vista matemático lo que se busca es estudiar

qué tipo de transformación debe aplicarse a una imagen para que el resultado sea otra imagen pero con mejores características visuales. Ecuación 1:

$$g(x, y) = T [f(x, y)] \quad (1)$$

Donde:

$f(x, y)$  representa la imagen de entrada

$g(x, y)$  es la imagen procesada

$T$  es un operador sobre  $f$ , la cual hace referencia a la región de pixeles por la que esté rodeada  $(x, y)$

La idea principal de establecer una vecindad alrededor de un pixel  $(x, y)$  es usar una subimagen cuyo centro sea  $(x, y)$ . El punto central de la subimagen va modificándose al recorrer la imagen principal desde la esquina superior izquierda hasta la esquina inferior derecha y el operador de transformación  $T$  se aplica a cada pixel  $(x, y)$  para generar una salida  $g(x, y)$ . La figura 1 muestra un ejemplo de un pixel cuya vecindad es de 3x3 y los pixeles vecinos están marcados con un símbolo (\*).

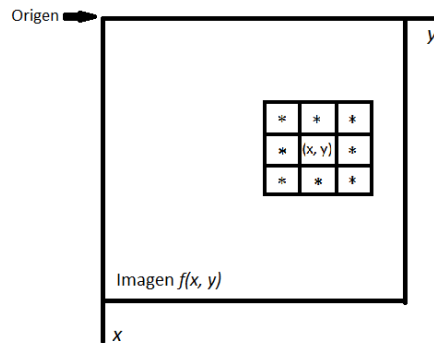


Figura 1: Pixel  $(x, y)$  con una vecindad de 3x3

## 2.2. Función de transformación

Diversos autores en la literatura hacen uso de la ecuación 2:

$$e_{i,j} = E^f_{i,j}(o_{i,j} - c * m^l_{i,j}) + (m^l_{i,j})^a \quad (2)$$

Donde:

$e_{i,j}$  representa la imagen mejorada

$o_{i,j}$  indica el valor de intensidad del  $(i, j)$ th pixel en la imagen de entrada

$m^l$  es la media local del  $(i, j)$ th pixel en un rango determinado por una ventana de tamaño  $w * w$  y está dada por la ecuación 3:



$$m^l = \frac{1}{w * w} \sum_{i=1}^w \sum_{j=1}^w o_{i,j} \quad (3)$$

$a, c$  son parámetros que generan variaciones de intensidad en los pixeles  $E^f_{i,j}$  denota la función de mejora dada por la ecuación 4:

$$E^f_{i,j} = \frac{k * m^g}{\sigma^l_{i,j} + b} \quad (4)$$

Donde:

$b, k$  son parámetros que generan variaciones de intensidad en los pixeles  $\sigma^l_{i,j}$  es la desviación estándar del  $(i, j)$ th pixel de la imagen de entrada en un rango determinado de tamaño  $w * w$   
 $m^g$  es la media global de la imagen y está dada por la fórmula 5:

$$m^g = \frac{1}{M * N} \sum_{i=1}^M \sum_{j=1}^N o_{i,j} \quad (5)$$

Donde:

$M$  Indica el tamaño de filas en la imagen de entrada

$N$  Indica el tamaño de columnas en la imagen de entrada

Esta función de transformación extiende el contraste de la imagen original centrada en su valor medio local, y tomando en cuenta información global. Los cuatro parámetros ( $a, b, c, k$ ), como ya se ha mencionado anteriormente, generan variaciones en los valores de intensidad y, por lo tanto, deben optimizarse para encontrar los más adecuados en la imagen a procesar.

### 2.3. Optimizador de Lobos Grises

El algoritmo Optimizador de Lobos Grises (Mirjalili y Lewis, 2014) está inspirado en su manera de buscar de forma óptima a sus presas para cazarlas. Una manada está formada entre 5 y 12 integrantes, los cuales se dividen en cuatro grupos basados en roles que cada lobo debe desempeñar para ayudar en el proceso de caza, así se imita la jerarquía de dominación de los lobos grises en la naturaleza. Los cuatro grupos son:

1. **Alfa.** Representa el líder, el cual es el responsable de tomar las decisiones más importantes acerca de la caza, el lugar donde dormir, a hora de despertar, etc..
2. **Beta.** Es el segundo nivel en la jerarquía. Los lobos beta son subordinados de los alfa y colaboran en la toma de decisiones, además cumplen las órdenes de los alfa. Juegan el papel de un asesor para el alfa y disciplinan al resto de la manada. Ellos son candidatos a convertirse en alfa cuando alguno de estos fallece o envejece.
3. **Delta.** La tercera categoría formada por deltas tiene que someterse a alfas y betas, pero domina a los omegas. Los lobos delta son generalmente centinelas, ancianos, cazadores y cuidadores.

4. **Omega.** En el orden más bajo de la jerarquía se encuentran los omegas. Suelen verse como niños de las crías dentro de la manada.

En el enfoque matemático de este algoritmo alfa representa la mejor solución, beta y delta las siguientes mejores soluciones. El resto de las soluciones son consideradas como omegas. Las fases que se llevan a cabo para conseguir su alimento son las siguientes:

1. Rodear a la presa. Los lobos grises rodean a la presa antes de cazarla. En el modelo matemático, la manada de lobos tiene que actualizar frecuentemente su posición (llamado valor de umbral) para ir acercándose a la presa mediante la fórmula 6:

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} * \vec{D} \quad (6)$$

Donde  $\vec{X}(t+1)$  hace referencia a la siguiente ubicación del lobo,  $\vec{X}_p(t)$  es el vector de posición de la presa,  $\vec{A}$  es un vector coeficientes y  $\vec{D}$  es un vector que depende de la ubicación actual de la presa  $\vec{X}_p(t)$ . Primeramente,  $\vec{D}$  es calculado por la ecuación 7:

$$\vec{D} = |\vec{C} * \vec{X}_p(t) - \vec{X}(t)|, \quad (7)$$

Donde  $\vec{X}(t)$  indica la posición actual del lobo gris y  $\vec{C}$  es, al igual que  $\vec{A}$ , un vector de coeficientes. Estos últimos son calculados de acuerdo a las ecuaciones 8 y 9:

$$\vec{A} = 2 * \vec{a} * \vec{r}_1 * \vec{a} \quad (8)$$

$$\vec{C} = 2 * \vec{r}_2 \quad (9)$$

Donde los componentes de  $\vec{a}$  son valores decrementados linealmente en el rango  $[2, 0]$  para las siguientes iteraciones y  $\vec{r}_1$  son vectores aleatorios en el intervalo  $[0, 1]$ . La ecuación para actualizar el parámetro  $\vec{a}$  es como se muestra en la ecuación 10, de acuerdo a ?):

$$\vec{a} = 2 - t * \frac{t}{T} \quad (10)$$

Donde  $t$  es la iteración actual y  $T$  es el número máximo de iteraciones.

De esta manera un lobo gris puede actualizar su posición respecto a la presa en cualquier ubicación.

2. Caza. En un espacio de búsqueda abstracto no se sabe la ubicación de la solución óptima (o presa), de manera que para simular matemáticamente el comportamiento de caza de los lobos grises se supone que alfa, beta y delta tienen el mejor conocimiento acerca de la ubicación potencial de la presa, esto se concluye debido a que son las 3 mejores soluciones hasta el momento. Por lo tanto, todos los agentes de búsqueda deben actualizar su posición respecto a la presa, de acuerdo a las posiciones de los 3 líderes, mediante 11, 12 y 13

$$\vec{D}_\alpha = |\vec{C}_1 * \vec{X}_\alpha - \vec{X}|, \vec{D}_\beta = |\vec{C}_2 * \vec{X}_\beta - \vec{X}|, \vec{D}_\delta = |\vec{C}_3 * \vec{X}_\delta - \vec{X}| \quad (11)$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 * \vec{D}_\alpha, \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 * \vec{D}_\beta, \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 * \vec{D}_\delta \quad (12)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (13)$$

3. Ataque. Para dar por terminada la tarea de caza, los lobos atacan a su presa. Matemáticamente, esta etapa es modelada mediante el decremento de  $\vec{a}$ , lo cual a su vez influye en  $\vec{A}$ , por lo que se puede notar que  $\vec{A}$  es un valor aleatorio en el intervalo  $[-2a, 2a]$  y hay que tener presente que  $\vec{a}$  decrementa en el transcurso de las iteraciones en el rango  $[2, 0]$ . Cuando los valores aleatorios de  $\vec{A}$  están en  $[-1, 1]$  la siguiente posición de un lobo gris puede darse en cualquier lugar entre la posición actual y la posición de la presa. Si  $|A| < 1$  entonces los lobos están obligados a atacar a la presa.
4. Búsqueda de la presa. Los lobos grises buscan de acuerdo a la posición de los líderes alfa, beta y delta. Se separan entre ellos para buscar presas y convergen cuando es necesario atacar. En el modelo matemático para representar la divergencia, se usa a  $\vec{A}$  con valores aleatorios mayores que 1 o menores que -1, así se obliga al agente de búsqueda a alejarse de la presa. Esta es la manera en que se enfatiza la exploración, permitiendo que el algoritmo haga una búsqueda global, pues cuando  $|A| > 1$  los lobos grises divergen de la presa con la posibilidad de encontrar una mejor.

### 3. Optimizador de Lobos Grises para Mejora de Imágenes Médicas

A continuación se describen las condiciones iniciales con las que se usó el algoritmo GWO.

#### 3.1. Inicialización de la población

La población inicial está formada por una matriz generada de manera aleatoria y con una distribución uniforme, la cual mediante sus filas representa a cada uno de los agentes de búsqueda y las columnas representan cada uno de los parámetros para ser optimizados. Para la generación aleatoria de estos parámetros es necesario considerar los límites de los valores, tanto superior como inferior. Según la literatura, estos toman valores de  $a \in [0, 1,5]$ ,  $b \in [0, 0,5]$ ,  $c \in [0, 1]$  y  $k \in [0,5, 1,5]$ , respectivamente.

#### 3.2. Cálculo de Función Objetivo

El criterio de evaluación de la imagen mejorada está basado en información de los bordes, tales como número de bordes, suma de la intensidad de los bordes

y la entropía global, además de la desviación estándar global, como se muestra en la fórmula 14:

$$F(e) = \sqrt[3]{\sigma_e * \log(\log(E(e^s))) * \frac{n_{edges}(e^s) * H(e)}{M * N}} \quad (14)$$

Donde:

$E(e^s)$  es la suma de todos los valores de intensidad de los pixeles detectados por Sobel en la imagen  $e^s$

$n_{edges}$  es el número total de pixeles en  $e^s$  cuyo valor de intensidad es mayor que un umbral definido

$H(e)$  representa la entropía global de la imagen mejorada y está dada por la ecuación 15:

$$H(e) = - \sum_{i=0}^{L_{max}-1} p_e(i) * \log_2(p_e(i)) \quad (15)$$

Donde:

$p_e(i)$  es la probabilidad de ocurrencia de pixeles con i-ésimo valor de intensidad en la imagen mejorada  $e$

$L_{max}$  es el valor máximo de intensidad presente en la imagen mejorada

### 3.3. Conjunto de datos

Para la implementación del algoritmo, 20 imágenes de tórax fueron tomadas de la base de datos perteneciente al National Institutes of Health Clinical Center (<https://clinicalcenter.nih.gov>), la cual contiene un total de 112120 imágenes de rayos X correspondientes a 30805 pacientes. Cada muestra consta de un archivo PNG con resolución de 1024x1024. Algunos de los metadatos que incluyen son: índice de imagen, etiquetas de búsqueda, número de seguimiento, ID del paciente, edad y sexo del paciente.

## 4. Resultados

En la Figura 2 se presentan 3 de las muestras usadas en el Algoritmo GWO.

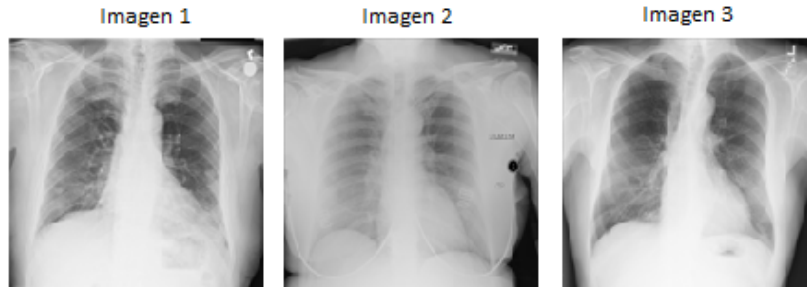


Figura 2: Imágenes usadas en GWO

En la Figura 3 se muestra el promedio de las imágenes de acuerdo al fitness obtenido a lo largo de las 60 generaciones. Se puede notar cómo el algoritmo incrementa el valor del fitness mientras el número de generaciones aumenta. Puede observarse que a partir de la iteración 25 el resultado no varía de manera significativa.

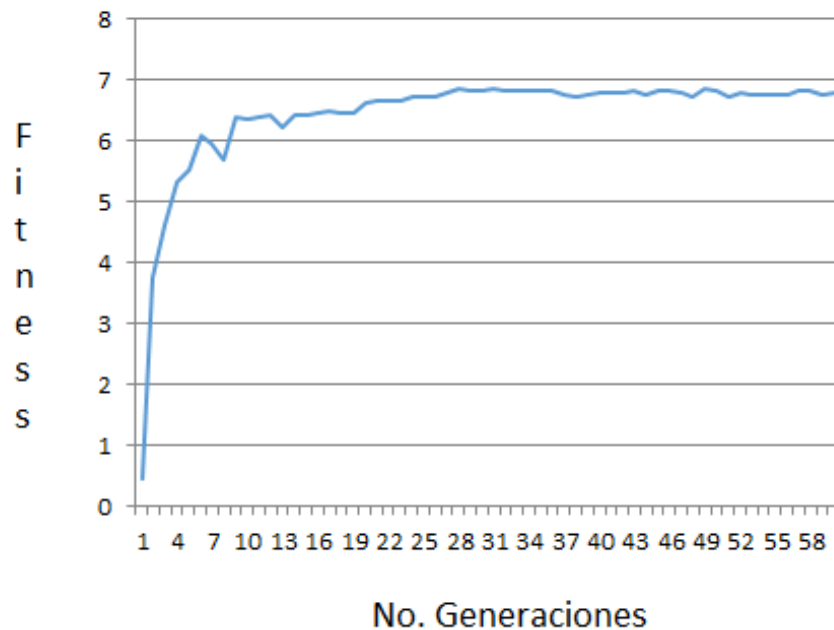


Figura 3: Promedio total del Fitness durante 60 generaciones de 40 individuos

Las imágenes resultantes de tórax fueron obtenidas mediante una fusión de suma y se hicieron las pruebas con la asignación de 4 pesos diferentes, tal como se muestra en la figura 4.

## 5. Conclusiones

La implementación del algoritmo Optimizador de Lobos Grises en la mejora de imágenes médicas se aplicó para buscar los parámetros óptimos, que ajustaran el valor de intensidad de los píxeles, dentro de la función de transformación para generar una imagen con características adecuadas para las tareas de procesamiento digital de imágenes, en este caso en el ambiente médico. Se notan algunas variaciones del valor fitness en el transcurso de las generaciones. Con la finalidad de conservar la diversidad de los individuos, a la vez que se hace una pequeña modificación en las soluciones candidatas se plantea la inclusión de operadores genéticos, los cuales mejoren el desempeño del algoritmo. Además es

necesario comparar el desempeño contra otros algoritmos de optimización para evaluar la eficacia entre ellos. Finalmente, pero no menos importante, cabe mencionar que en el procedimiento original de GWO cuando se actualizan las posiciones de los agentes de búsqueda, si el líder alfa es reemplazado éste se pierde sin tener oportunidad de sustituir a algún otro agente con menor jerarquía, por lo cual se considera necesario el hecho de modificar esa tarea, para no excluir una buena solución candidata.

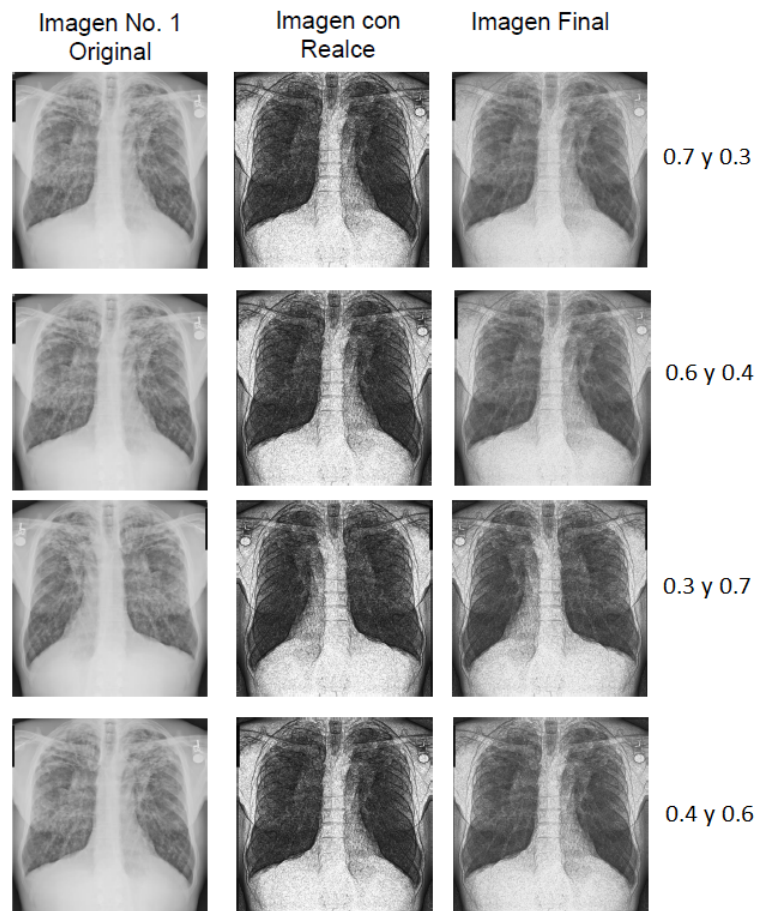


Figura 4: Imágenes obtenidas a partir de las originales y de GWO

## Referencias

- Chaira, T. (2014). "An improved medical image enhancement scheme using type ii fuzzy set", *Applied soft computing*, vol. 25, pp. 293–308.
- Daniel, E. and Jude, A. (2015). Optimum green plane masking for the contrast enhancement of retinal images using enhanced genetic algorithm. 126.
- Dhal, K. G. and Das, S. (2017). "Cuckoo search with search strategies and proper objective function for brightness preserving image enhancement", *Pattern Recognition and Image Analysis*, vol. 27, pp. 695–712.
- Dos Santos Coelho, L., Sauer, J. G., and Rudek, M. (2009). "Differential evolution optimization combined with chaotic sequences for image contrast enhancement", *Chaos, Solitons & Fractals*, vol. 42, pp. 522–529.
- Gao, H. and Zeng, W. (2015). "Color image enhancement based on ant colony optimization algorithm", *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 13, pp. 155–163.
- Hoseini, P. and Shayesteh, M. G. (2013). "Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing", *Digital Signal Processing*, vol. 23, pp. 879–893.
- Kumar, S., Pant, M., and Ray, A. K. (2014). "De-ie: differential evolution for color image enhancement", *International Journal of System Assurance Engineering and Management*, pp. 1–12.
- Long, M., Li, Z., Xie, X., Li, G., and Wang, Z. (2018). "Adaptive image enhancement based on guide image and fraction-power transformation for wireless capsule endoscopy", *IEEE transactions on biomedical circuits and systems*, vol.12, pp. 993–1003.
- Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). "Grey wolf optimizer. Advances in engineering software", vol. 69, pp. 46–61.
- Nickfarjam, A. M. and Ebrahimpour-Komleh, H. (2017). "Multi-resolution gray-level image enhancement using particle swarm optimization", *Applied Intelligence*, vol. 47, pp. 1132–1143.
- Shanmugavadivu, P., Balasubramanian, K., and Muruganandam, A. (2014). "Particle swarm optimized bi-histogram equalization for contrast enhancement and brightness preservation of images", *The Visual Computer*, vol. 30, pp. 387–399.
- Sonali, Sahu, S., Singh, A. K., Ghrera, S., and Elhoseny, M. (2019). "An approach for denoising and contrast enhancement of retinal fundus image using clahe", *Optics Laser Technology*, vol. 110, pp. 87 – 98. Special Issue: Optical Imaging for Extreme Environment.
- Ye, Z., Wang, M., Hu, Z., and Liu, W. (2015). "An adaptive image enhancement technique by combining cuckoo search and particle swarm optimization algorithm", *Computational Intelligence and Neuroscience*.

# Capítulo 21

## Transformada de Fourier en tiempo corto para el análisis de espectrogramas

Ricardo Ramos Aguilar, José Arturo Olvera López e Ivan Olmos Pineda

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación  
Doctorado en Ingeniería del Lenguaje y del Conocimiento  
ricramosa1@gmail.com, {aolvera, iolmos}@cs.buap.mx

**Resumen.** Uno de los problemas más comunes en la implementación de la transformada de Fourier de tiempo corto es la definición de parámetros, debido a que estos modifican al espectrograma en tiempo o frecuencia. En un proceso de extracción de características cambia las que se obtienen, por tal motivo se debe conocer como afecta a un espectrograma bajo la modificación de ciertos parámetros. En este artículo se muestra un análisis de los parámetros que influyen en la calidad de resolución tiempo-frecuencia de un espectrograma, específicamente, para el tamaño de la ventana y traslape. Se muestran resultados para una señal periódica para diferentes tipos y tamaños de ventana. Por otro lado, se hicieron unas pruebas con las señales EEG, por medio de diferentes ventanas, tamaños y porcentaje de traslape; estos se ajustaron para la base de datos de ataque epilépticos de la Universidad de Bonn, mostrando los resultados por medio de un análisis visual de los espectrogramas.

**Palabras clave:** Señales EEG, transformada de Fourier de tiempo corto, espectrograma, epilepsia

### 1. Introducción

La transformada de Fourier de tiempo corto (STFT) es una herramienta utilizada comúnmente para el procesamiento de señales, en áreas como: el procesamiento de imágenes, procesamiento de voz, biología y medicina Mateo y Talavera (2018) y una aplicación con resultados competitivos con respecto a otros métodos es en la identificación de bioseñales como las Electrocardiográficas (ECG) y Electroencefalográficas (EEG) (Tzallas et al., 2009); una aplicación común es la médica donde se pretende identificar diferentes enfermedades, procesos cognitivos o para la aplicación de interfaces cerebro computadora, sin embargo, los resultados obtenidos dependen en gran parte de la etapa de extracción de características, las cual es aplicada a los espectrogramas y estos a su vez están relacionados con la selección de parámetros. Una alternativa a la STFT es la transformada Wavelet debido a los resultados competitivos que también ha mostrado. Un estudio realizado por (Kiyimik et al., 2005) menciona que WT tiene



mejores resultados con respecto a resolución y mejor desempeño de visualización en el mapeo(imagen) resultante de señales EEG en ataques epilépticos, sin embargo, en ese estudio se clasificaron esas señales y se mostró que STFT se desempeña mejor con redes neuronales y también que consume menos tiempo. Con un resultado similar para señales del mismo tipo, (Subha et al., 2010) menciona que STFT procesa bien las señales debido a su bajo costo computacional y que para aplicaciones reales es la mejor opción, sin embargo, si no se necesitara una aplicación que se desempeñe en tiempo real WT podría ser una mejor opción, por su desempeño de evaluación de la señales.

Por otra parte, el trabajo de (Allen y MacKinnon, 2010) compara métodos como filtros, transformada de Wavelet continua (CWT) y STFT, sobre distintas señales, genéricas y EEG's. En el que para STFT usa una ventana Gaussiana como parámetro y se demuestra que STFT es superior en términos de resolución temporal y espectral.

De los trabajos revisados, se puede concluir que no hay una distribución que sea mejor, sin embargo, se puede deducir que los resultados de un análisis de señales, en particular para EEG, aplicando STFT depende del tamaño de la ventana en gran parte, debido a que es la desventaja de esta distribución.

Este trabajo muestra algunos de los resultados visuales en donde parámetros como el ventaneo son importantes para la resolución de un espectrograma, también se muestra que no es el único. La sección siguiente, presenta la metodología; en la sección 3 se describen los resultados obtenidos; finalmente, en la sección 4 se muestran las conclusiones.

## 2. Método experimental

En esta sección, se muestra la teoría utilizada para el desarrollo de la metodología.

### 2.1. Transformada de Fourier de tiempo corto

En aplicaciones prácticas, la STFT es implementada como una ventana deslizante ajustada a una señal. A partir de  $x[t]$  una señal de duración  $t$ , segmentos de datos se extraen en intervalos regulares usando una ventana de tiempo-limitado  $w[n]$  Goodwin (2008); los segmentos(frames) de la señal pueden ser expresados como:

$$x_l[n] = w[n]x[x + lL], \quad 0 \leq n \leq N - 1 \quad (1)$$

donde  $N$  es la longitud de la ventana,  $l$  es el índice de ventana y  $L$  es el tamaño de salto, es decir el espaciado en muestras entre aplicaciones consecutivas de la ventana de extracción deslizante; el índice  $n$  es un índice de tiempo local, es un índice relativo al comienzo del desplazamiento de la ventana. Para cada frame de la señal se aplica la transformada de Fourier discreta (DFT) Goodwin

(2008), quedando de la siguiente forma

$$X[k, l] = \sum_{n=0}^{N-1} x_l[n] e^{-i2\pi nk/K} = \sum_{n=0}^{N-1} w[n] x[xn + lL] e^{-i2\pi nk/K}, \quad (2)$$

donde  $K$  es el tamaño de la DFT y  $k$  es un índice de frecuencia o índice bin. La STFT  $X[k, l]$  entonces caracteriza el comportamiento tiempo-frecuencia local de la señal al rededor del tiempo  $lL$  y el bin  $k$ ; para una tasa de muestreo  $F_s$ , estos índices discretos corresponde al tiempo continuo  $lL/F_s$  y frecuencia  $kF_s/K$  Goodwin (2008). Para simplificar la notación, un frecuencia radial

$$\omega_k = 2\pi k/K \quad (3)$$

frecuentemente se incorpora, entonces la expresión STFT se convierte en

$$X[k, l] = \sum_{n=0}^{N-1} w[n] x[xn + lL] e^{-i\omega_k n}, \quad (4)$$

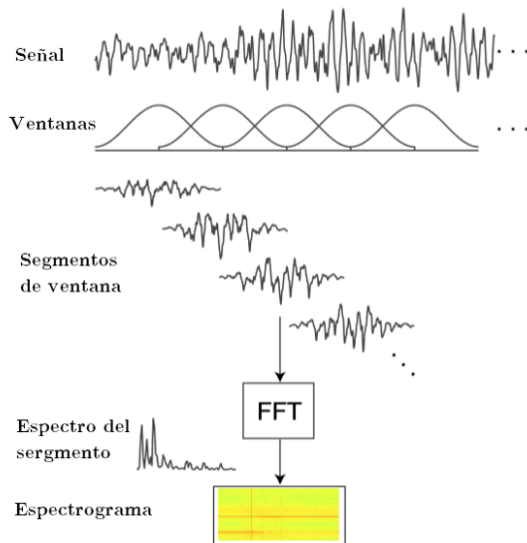
Tomando en cuenta la ventana a lo largo del tiempo, la STFT resulta como una representación espectral de un segmento de tiempo de la señal de entrada; interpretando  $X[k, l]$  como una función de la frecuencia  $k$  para cada valor del índice de tiempo  $l$ , la STFT corresponde a una serie de espectros localizados en el tiempo. Alternativamente, la STFT puede verse como una función de tiempo para cada frecuencia; interpretando  $X[k, l]$  como una serie de tiempo que es una función de  $l$  para cada bin (puntos para calcular la DFT)  $k$ , la STFT corresponde a un banco de filtros que descompone la señal de entrada en canales o sub-bandas de frecuencia Goodwin (2008).

La STFT se puede interpretar como las transformada de Fourier de una secuencia ventaneada o como una operación de filtrado lineal. Debido a la función de ventana de la que depende, se puede obtener un localización en tiempo de la señal también. Este método es usado para señales no estacionarias como señales de voz (Prabhu, 2018), o en nuestro caso como en las señales EEG. En un principio, se aplicó la STFT a cada una de las señales de las bases de datos; ya que como se mencionó este método es usado para el análisis/síntesis de señales.

Para la aplicación de la STFT se deben tener en cuenta diferentes parámetros que de manera general e implícita son mostrados en la figura 1, los cuales son:

- Tipo y tamaño de ventana
- Ancho del lóbulo principal
- Tamaño de la FFT
- Traslape

Estos parámetros son ajustados comúnmente con la finalidad de tener un ajuste medio entre resolución en tiempo y frecuencia. Sin embargo, también pueden ser ajustados a nuestra conveniencia, si se busca alguna frecuencia como las que se mueven las ondas de EEG o actividad que se genera durante periodos de tiempo específicos.



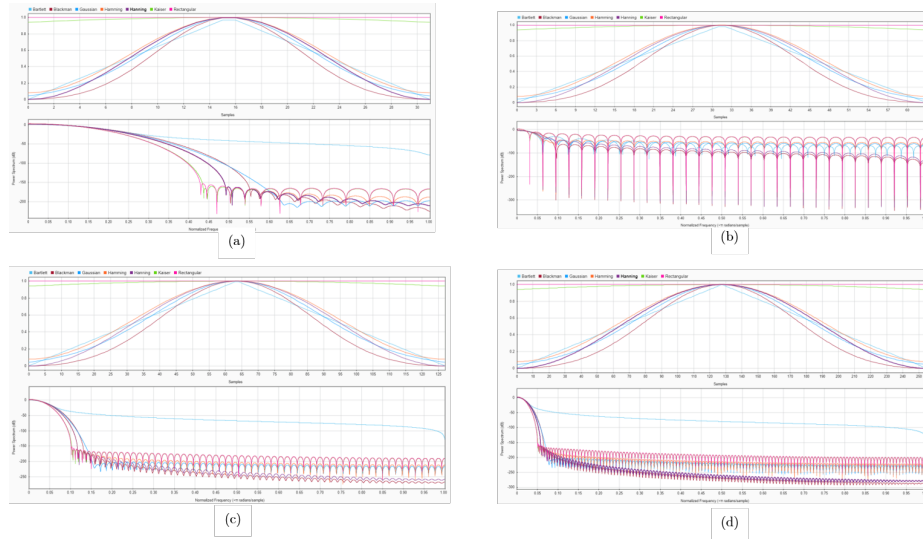
**Figura 1.** Proceso de la STFT. Imagen tomada de Sethares (2009)

En distintos trabajos se aplican la STFT para el análisis de señales EEG, sin embargo, la mayoría no muestran o no justifican los parámetros usados en la transformada de Fourier en tiempo corto; diferentes autores han propuesto algunos con objetivos diferentes (detección de enfermedades, estados mentales, procesos cognitivos, entre otros) Gerla et al. (2018), Allen y MacKinnon (2010), Tzallas et al. (2009), Duque-Munoz et al. (2014), Van Hoey et al. (1997) y Zabidi et al. (2012) que se muestran en la tabla 1, la mayoría de ellos usan ventanas de tipo Hanning, Hamming y Gaussiana, con distintos tipos de duración; sus parámetros son obtenidos basados en la experimentación para obtener los óptimos. Sólo Duque-Munoz et al. (2014) justifica su ventaneo con un método basado en una función de tapering para localizar tiempos y ajustarlos a la ventana D Marinez-Vargas et al. (2012); a pesar de que existen trabajos que han usado STFT ellos no mencionan sus parámetros, por lo tanto, no tienen que justificarlos. Axel Robel un reconocido investigador de París que trabaja en el procesamiento digital de señales, dice que no hay una duración de ventana óptima, todo depende de lo que se esté buscando. Carlos Mateo en Mateo y Talavera (2018) menciona que existen métodos para el ventaneo como STFT adaptativo o multi-resolución con diferentes ventajas y desventajas como la complejidad computacional, al hacer uso de bancos de filtros o la dependencia de características de las señales; en el trabajo de Mateo y Talavera (2018) los autores proponen usar ventanas de diferentes tamaños para diferentes frecuencias, el método fue probado con señales genéricas y con señales electrocardiográficas (ECG) concluyendo que se tiene una buena resolución tiempo-frecuencia simultáneamente.

**Tabla 1.** Parámetros de ventanas en trabajos relacionados con EEG

Referencia	Artículo	Ventana
Tzallas A., 2009	Epileptic Seizure Detection in EEGs using Time-Frequency Analysis	Hamming a 64 muestras.
Nicoale C., 2011	Study of the effects of the short time Fourier transform configuration on EEG spectral estimates	Hamming, Gaussian, Hann y Bartlett-Hann a 128, 256, 512 y 1024 muestras y 96% traslape.
David P., 2010	Time-frequency analysis of movement-related spectral power in EEG during repetitive movements: a comparison of methods	Gaussiana y Hamming a 64 y 128 muestras.
Van G., et al., 1997	Time-Frequency Analysis of EEG Signals	1 s.
Gerla V., et al., 2018	Comparison of short time Fourier transform and continuous wavelet transform for frequency analysis of sleep EEG	60 s. y 1 a 3 s.
Duque L., et al., 2014	Identification and monitoring of brain activity based on stochastic relevance analysis of short-time EEG rhythms	Gaussian window a 2.9 s (503 muestras) con 40% de traslape
Zabidi A., et al., 2012	Short-time Fourier Transform Analysis of EEG Signal Generated During Imagined Writing	Hamming de 0.1 a 0.8s.

### 3. Resultados



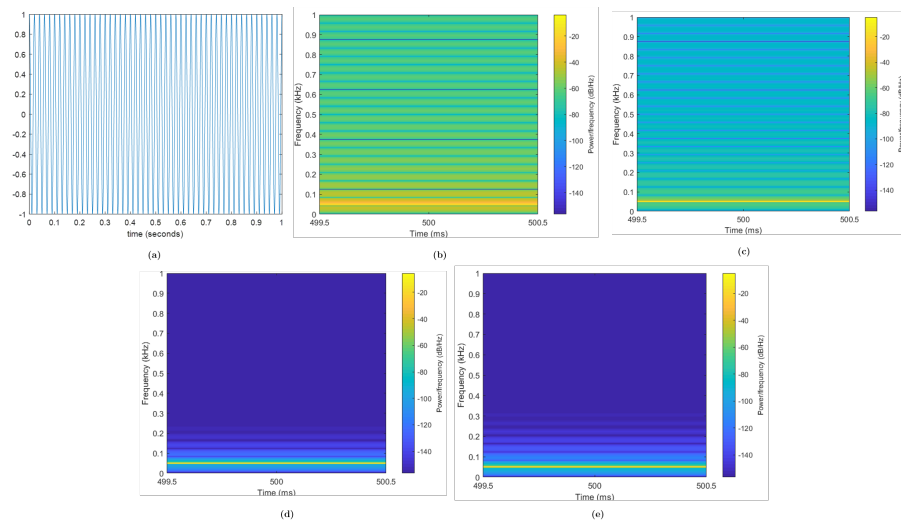
**Figura 2.** Ventanas (rojo-Blackman, azul-Gaussiana, anajaranjado-Hamming, morado-Hanning, verde-Kaiser y rosa-rectangular) a diferentes longitudes (32, 64, 128 y 512) y sus respectivos espectros.

Tomando en cuenta un análisis visual sobre lóbulos secundarios y principal, podemos seleccionar una ventana como opción aceptable, debido a que un lóbulo más delgado muestra frecuencias cercanas a la de la señal; por otro lado, los lóbulos secundarios, entre más pequeños, muestran una menor dispersión de energía,

debido a que la mayor parte está concentrada en el principal, sin embargo, estos parámetros no son los únicos que se pueden contemplar para el análisis de una ventana (Prabhu, 2018), aunque, posiblemente, si es suficiente para un análisis de espectrogramas basado en picos espectrales, por lo tanto, al tomar en cuenta el lóbulo principal de la ventana Hanning como el tercer más delgado, muy cercano a los dos primeros y los lóbulos segundos pequeños antes de Blackman, a partir de estos parámetros, Hanning representa una buena opción.

#### 4. Resultados experimentales

Para los experimentos, se tomaron en cuenta cuatro ventanas, Gaussiana por ser la cercana a la ideal, la rectangular por ser la ideal en el tiempo, Blackman al tener los lóbulos más pequeños y Hanning por mantener un buena relación entre ancho mínimo del lóbulo principal y altura de lóbulos secundarios.



**Figura 3.** (a) Señal coseno a 50 Hz a una frecuencia de muestreo de 2000 Hz , con sus espectrogramas a partir de una ventana (b) rectangular, (c) Gaussiana, (d) Blackman y (e) Hanning.

Después de un análisis general de los parámetros, se notó el comportamiento de los espectrogramas, los cuales son afectados por los diferentes parámetros. En esta sección se pretende hacer un análisis de los espectrogramas para las señales EEG del conjunto de datos de epilepsia, específicamente para las clases de personas sanas y pacientes con ataques epilépticos. Algunos de estos espectrogramas se utilizaron para la extracción de características, con los cuales se nota como cambian las características por medio de algoritmos de clasificación.

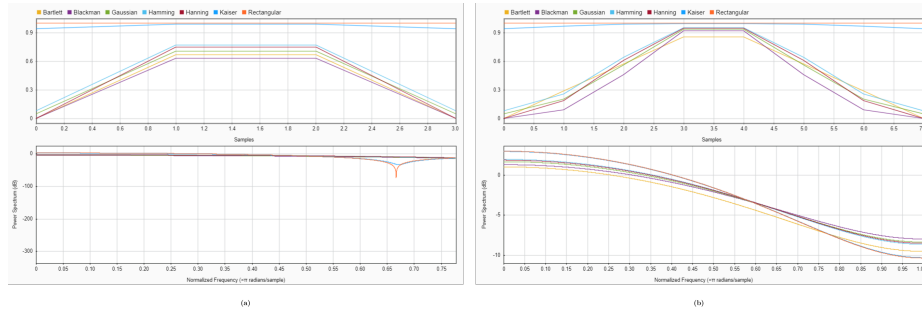


Figura 4. ventanas de longitud pequeña, (a) 4 y (b) 8 muestras.

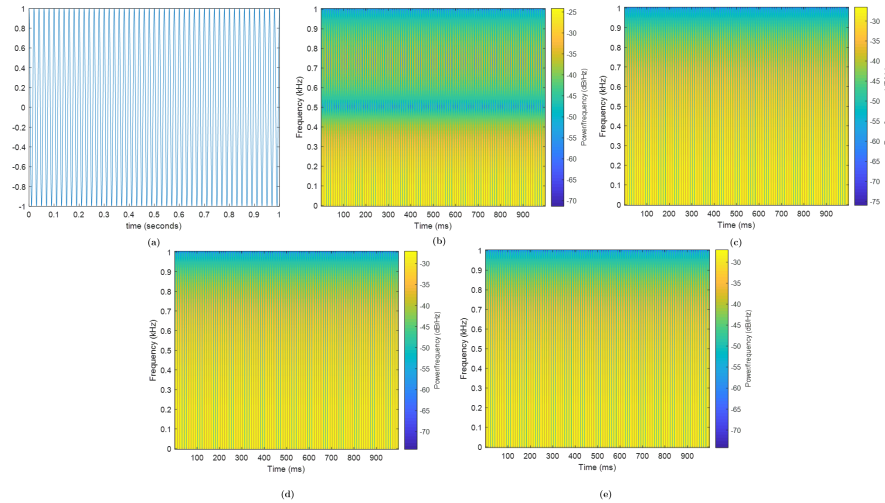
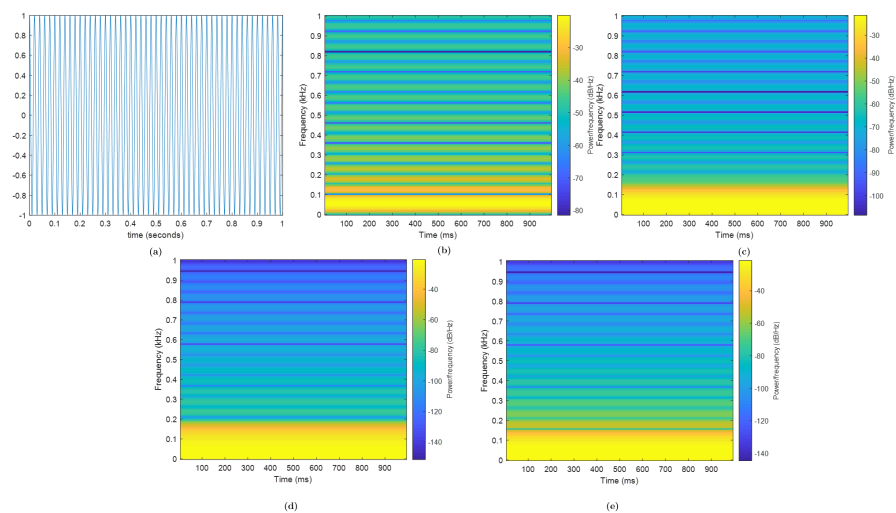


Figura 5. (a) Señal coseno a 50 Hz , con sus espectrogramas a partir de 500 ventanas (b) rectangular, (c) Gaussiana, (d) Blackman y (e) Hanning.

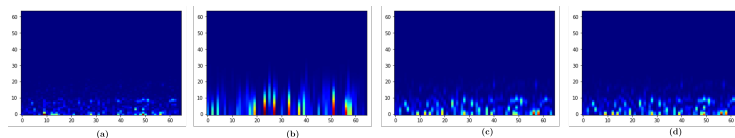
#### 4.1. Señales EEG

Al comparar los espectros de la figura 7 con la figura 8, si nota un alargamiento de la energía sólo sobre el eje horizontal (tiempo), aunque, realmente lo que sucede es que se enfoca más a la frecuencias mostradas en el eje vertical. Algo muy similar a colocar una sola ventana, que sólo nos muestra frecuencia, y cuando una ventana es muy angosta, nos da cambios a lo largo del tiempo.

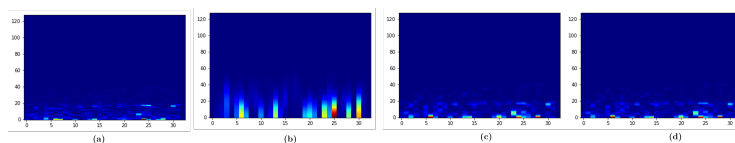
El traslape implica un enfoque más general a la señal, entre más porcentaje exista, tiene a parecerse una sola ventana, con la diferencia de ser un análisis individual, por ventana. Para los ocho espectros, de las figuras 9 y 10 son muy parecidos a los ocho anteriores, pero el traslape adelgaza la energía mostrada a lo largo del tiempo.



**Figura 6.** (a) Señal coseno a 50 Hz , con sus espectrogramas a partir de 500 ventanas con 90 % de traslape (b) rectangular, (c) Gaussiana, (d) Blackman y (e) Hanning.

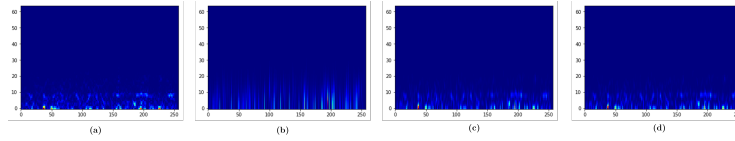


**Figura 7.** Espectrogramas de las señales EEG de una persona en estado relajado con los ojos abiertos, para ventanas de tamaño 64 sin traslape de tipo (a) Rectangular, (b) Gaussiana , (c) Blackman y (d) Hanning.

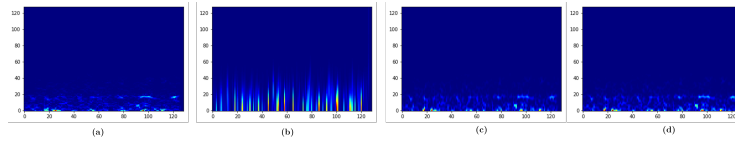


**Figura 8.** Espectrogramas de las señales EEG de una persona en estado relajado con los ojos abiertos, para ventanas de tamaño 128 sin traslape de tipo (a) Rectangular, (b) Gaussiana , (c) Blackman y (d) Hanning.

Si el traslape es mayor, la energía parece reducirse a lo largo del eje del x (tiempo), y en términos de resolución tiempo-frecuencia, una ventana grande enfocada para las frecuencia, tendría un mejor resolución con un traslape de mayor porcentaje.



**Figura 9.** Espectrogramas de las señales EEG de una persona en estado relajado con los ojos abiertos, para ventanas de tamaño 64 con 75 % traslape de tipo (a) Rectangular, (b) Gaussiana , (c) Blackman y (d) Hanning.



**Figura 10.** Espectrogramas de las señales EEG de una persona en estado relajado con los ojos abiertos, para ventanas de tamaño 128 con 75 % traslape de tipo (a) Rectangular, (b) Gaussiana , (c) Blackman y (d) Hanning.

## 5. Conclusiones y trabajo a futuro

De los trabajos analizados se notó que no muestran los parámetros con respecto a la STFT, algunos de los que muestran no los están justificando. De un análisis teórico breve y unos experimentos, se observa que los parámetros son importantes, ya que el espectrograma se ve afectado; particularmente, debemos de tomar en cuenta, ancho de la ventana, traslape y frecuencia de muestreo de la señal, el tipo de ventana puede ser Hanning, Hamming o Gaussiana.

Debido a la energía que se muestra en los espectrogramas, probablemente se debe proponer una ventana para cada tipo de clase, ya que si se muestra energía a lo largo del tiempo o frecuencia, las ventanas deben de tener diferente longitud.

Como trabajo a futuro, se pretende usar las ventanas que muestren una mejor representación en términos de energía con respecto al lóbulo principal del tipo de ventana y aplicarlo en una etapa de extracción de características.

La experimentación en este artículo para diferentes parámetros de la STFT es muy general, se pretende analizar las ventanas, con respecto a espectro, energía, lóbulos, entre otras.



## Referencias

- Allen, D. P., y MacKinnon, C. D. (2010). Time-frequency analysis of movement-related spectral power in eeg during repetitive movements: A comparison of methods. *Journal of Neuroscience Methods*, 186(1), 107-115.
- D Marinez-Vargas, J., Castellanos I Godino-Lorenzo, J., y Castellanos-Dominguez, G. (2012). Time-frequency based feature selection for discrimination of non-stationary biosignals. *EURSIP Journal on Advances in signal Preocessing*(1), 2019.
- Duque-Munoz, L., Espinosa-Oviedo, J. J., y Castellanos-Dominguez, C. G. (2014, 28 de Aug). Identification and monitoring of brain activity based on stochastic relevance analysis of short-time eeg rhythms. *BioMedical Engineering OnLine*, 13(1), 123.
- Gerla, V., Saifutdinova, E., Macas, M., Mladek, A., y Lhotska, L. (2018). P01-comparison of short-time fourier transform and continuous wavelet transform for frequency analysis of sleep eeg. *Clinical Neurophysiology*, 129(4), e14.
- Goodwin, M. M. (2008). The stft, sinusoidal models, and speech modification. En J. Benesty, M. M. Sondhi, y Y. A. Huang (Eds.), *Springer handbook of speech processing* (pp. 229–258). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Kiyimik, M., Guler, I., Dizibuyuk, A., y Akin, M. (2005). Comparison of stft and wavelet transform methods in determining epileptic seizure activity in eeg signals for real-time application. *Computers in Biology and Medicine*, 35(7), 603 - 616.
- Mateo, C., y Talavera, J. A. (2018). Short-time fourier transform with the window size fixed in the frequency domain. *Digital Signal Processing*, 77, 13 - 21. (Digital Signal Processing and SoftwareX - Joint Special Issue on Reproducible Research in Signal Processing)
- Prabhu, K. (2018). *Window functions and their applications in signal processing*. CRC Press.
- Sethares, W. (2009, 01). Rhythm and transforms, perception and mathematics. En (p. 1-10). doi: 10.1007/978-3-642-04579-0\_1
- Subha, D. P., Joseph, P. K., Acharya U, R., y Lim, C. M. (2010, 01 de Apr). Eeg signal analysis: A survey. *Journal of Medical Systems*, 34(2), 195–212.
- Tzallas, A. T., Tsipouras, M. G., y Fotiadis, D. I. (2009). Epileptic seizure detection in eegs using time-frenquecy analysis. *IEEE Transactions on Information Technology in Biomedicine*, 13(5), 703-710.
- Van Hoey, G., Philips, W., y Lemahieu, I. (1997). Time-frequency analysis of eeg signals. En *Proceedings of the prorisc workshop on circuits, systems and signal processing*.
- Zabidi, A., Mansor, W., Lee, Y. K., y Fadzal, C. W. N. F. C. W. (2012, Sept). Short-time fourier transform analysis of eeg signal generated during imagined writing. En *2012 international conference on system engineering and technology (icset)* (p. 1-4).

## Índice de Autores

<b>Nombre del Autor</b>	<b>Nacionalidad</b>	
Adelina Escobar Acevedo	Mexicana	
Aida Anaí Aparicio Arroyo	Mexicana	
Alfredo García Suárez	Mexicana	
Amparo Dora Palomino Merino	Mexicana	
Arturo Fernández Téllez	Mexicana	
Carlos Guillén Galvan	Mexicana	
Carlos Leopoldo Carreón Díaz de León	Mexicana	
Cristina López Ramírez	Mexicana	
Daniel Marcelo González Arriaga	Mexicana	
Daniel Sánchez Ruiz	Mexicana	
Darnes Vilariño Ayala	Mexicana	Compilador
David Céspedes Hernández	Mexicana	
Eustolia Carreón Esteban	Mexicana	
Guillermo De Ita Luna	Mexicana	
Guillermo Tejeda Muñoz	Mexicana	
Ivan Olmos Pineda	Mexicana	
José Arturo Olvera López	Mexicana	
José de Jesús Lavalle	Mexicana	
José Fernando Reyes Cortés	Mexicana	
Josefina Guerrero García	Mexicana	
Juan Manuel González	Mexicana	
Julio Jesús Salas Conde	Mexicana	
Lauro Reyes Cocoltzi	Mexicana	
Manuel Martín Ortíz	Mexicana	
María Aurora Diozcora Vargas Treviño	Mexicana	
María Josefa Somodevilla García	Mexicana	Compilador
Miguel Angel Ortega Palacios	Mexicana	
Miguel Rodríguez Hernández	Mexicana	
Pérez Camacho Blanca Nydia	Mexicana	
Ricardo Ramos Aguilar	Mexicana	
Sergio Vergara Limon	Mexicana	
Soraia Prietch	Brasileña	
Stephanie Vázquez González	Mexicana	
Tonatiuh García Chávez	Mexicana	
Víctor Manuel Morales de Jesús	Mexicana	
Yesenia Nohemí González Meneses	Mexicana	
Yolanda Moyao	Mexicana	
Yuridiana Alemán Muñoz	Mexicana	

## Compiladores

Mireya Tovar Vidal  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
María Josefa Somodevilla García

## Revisores

Amparo Dora Palomino Merino  
Arturo Olvera López  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
David Pinto Avendaño  
Fernando Reyes Cortes  
Guillermo De Ita Luna  
Hilda Castillo Zacatelco  
Iván Olmos Pineda

José Luis Carballido Carranza  
Josefina Guerrero García  
Juan Manuel González Calleros  
Manuel Martín Ortíz  
María Aurora Diozcora Vargas Treviño  
María Josefa Somodevilla García  
Mireya Tovar Vidal  
Sergio Vergara Limon

## Editores

Mireya Tovar Vidal  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
María Josefa Somodevilla García

Progresos actuales en la Ingeniería del Lenguaje y del Conocimiento  
Coordinado por  
Mireya Tovar Vidal  
Claudia Zepeda Cortés  
Darnes Vilariño Ayala  
María Josefa Somodevilla García  
está a disposición en html en la página  
de la Facultad de Ciencias de la Computación  
de la Benemérita Universidad Autónoma de Puebla (BUAP)  
<https://www.cs.buap.mx/~mtovar/doc/Libros/LibroCD20.pdf>  
a partir de diciembre de 2020  
Peso del archivo: 33.5 MB