EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

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

Research in Computing Science

Vol. 153 No. 7
July 2024

Research in Computing Science

Research in Computing Science, Año 23, Volumen 153, No. 7, julio de 2024, es una publicación mensual, editada por el Instituto Politécnico Nacional, a través del Centro de Investigación en Computación. Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, Ciudad de México, Tel. 57 29 60 00, ext. 56571. https://www.rcs.cic.ipn.mx. Editor responsable: Dr. Grigori Sidorov. Reserva de Derechos al Uso Exclusivo del Título No. 04-2019-082310242100-203. ISSN: en trámite, ambos otorgados por el Instituto Politécnico Nacional de Derecho de Autor. Responsable de la última actualización de este número: el Centro de Investigación en Computación, Dr. Grigori Sidorov, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othon de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738. Fecha de última modificación 01 de julio de 2024.

Research in Computing Science, year 23, Volume 153, No. 7, July 2024, is published monthly by the Center for Computing Research of IPN.

Volume 153(7)

Advances in Artificial Intelligence

Iris Iddaly Méndez-Gurrola (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2024

ISSN: in process

Formerly ISSNs: 1870-4069, 1665-9899

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

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

Indexed in LATINDEX, DBLP and Periodica

Electronic edition

ISSN 1870-4069

Table of Contents

Page

ISSN 1870-4069

ELT: Transformadores para la comprensión de la lengua de señas mexicana a través del preentrenamiento de puntos de referencia en imágenes

Víctor Martínez-Sánchez[1], Iván Villalón-Turrubiates[1], Francisco Cervantes-Álvarez[1], Carlos Hernández-Mejía[2], Delia Torres-Muñoz[3]

[1] Instituto Tecnológico de Estudios Superiores de Occidente, Guadalajara, México

[2] Tecnológico Nacional de México, Instituto Tecnológico Superior de Misantla, México

[3] Instituto Tecnológico Superior de San Martín Texmelucan, México

{ng683728, villalon, fcervantes}@iteso.mx, {cmahernandez, deletsm}@gmail.com

Resumen. Un nuevo modelo de representación de la Lengua de Señas Mexicana llamado Encoded Landmarks from Transformers (ELT) es introducido. ETL está basado en codificadores bidireccionales preentrenados usando puntos de referencia en imágenes no etiquetadas mediante un vector enmascarado en todas las capas. En consecuencia, el modelo ETL preentrenado puede ajustarse finamente con solo una capa de salida adicional con el propósito de generar modelos de clasificación y subtitulación de secuenciación de imágenes. El rendimiento es evaluado mediante el conjunto de datos MX-ITESO-100. ELT evidencia una ganancia de precisión real del 3 % comparado contra un modelo tradicional Long Short-Term Memory (LSTM) bidireccional. Adicionalmente, el modelo ELT reduce el tiempo de entrenamiento hasta en un 28 % y permite realizar un ajuste fino en un tiempo altamente competitivo.

Palabras clave: ETL, transformadores, lengua de señas mexicana.

ELT: Transformers for the Understanding of Mexican Sign Language through Pre-training of Reference Points in Images

Abstract. A new representation model of Mexican Sign Language called Encoded Landmarks from Transformers (ELT) is introduced. ETL is based on pre-trained bidirectional encoders using landmarks on unlabeled images using a masked vector on all layers. Consequently, the pre-trained ETL model can be fine-tuned with only one additional output layer for the purpose of generating image sequencing classification and captioning models. The performance is

evaluated using the MX-ITESO-100 data set. ELT shows a real precision gain of 3% compared to a traditional bidirectional Long Short-Term Memory (LSTM) model. Additionally, the ELT model reduces training time by up to 28% and allows fine tuning to be carried out in a highly competitive time.

Keywords: ETL, transformers, Mexican sign language.

1. Introducción

La Lengua de Señas Mexicana (LSM) es la lengua de señas utilizada por la comunidad sorda en México. Ha sido oficialmente reconocida como un idioma en nuestro país desde 2003, según la Ley General para la Inclusión de Personas con Discapacidad. La LSM sigue reglas particulares gramaticales y léxicas; siendo una herramienta crucial para la comunicación entre la población sorda en México. Por esta razón, es imperativo contar con tecnología que facilite la integración de las personas sordas en la sociedad mexicana.

Los nuevos enfoques basados en redes de transformadores prometen resultados favorables para el procesamiento del lenguaje natural y por lo tanto vincular la tendencia de desarrollo en transformadores con los avances en la LSM. Esta investigación propone vectorizar la representación de un gesto o seña utilizando puntos de referencia de una o ambas manos como un componente fundamental de la propuesta. De manera paralela, el uso del modelo preentrenado ETL permite agilizar el entrenamiento de modelos para otras investigaciones a través del ajuste fino.

2. Trabajos relacionados

El concepto de representación vectorial para unidades lingüísticas, específicamente palabras o lexemas, ha sido un tema destacado en el Natural Language Processing (NLP). La utilización de vectores continuos permite capturar matices semánticas y relaciones dentro del espacio lingüístico. El algoritrmo word2vec [11] ha demostrado eficacia en la extracción de características; logrando que las incrustaciones de palabras capturen relaciones semánticas entre palabras. Matthew E. [12] presenta un novedoso modelo de representación de palabras llamado Embeddings from Language Models (ELMo). ELMo extiende la idea mediante la introducción de incrustaciones contextualizadas que permiten una representación más matizada de las palabras basada en el contexto en donde aparecen.

De acuerdo a los avances más reciente en NLP, existe en la literatura dos conceptos fundamentales: modelos preentrenados [14] y la arquitectura de transformadores [16]. Alec Radford [13] propone un enfoque innovador conocido como Generative Pre-training (GPT) en donde es posible destacar que la fase de preentrenamiento capacita al modelo con una comprensión integral del lenguaje y por lo tanto permite capturar información contextual y adquirir conocimiento de estructuras sintácticas y semánticas. De forma semejante a el modelo de lenguaje GPT, Jacob Develin [5] introduce un modelo llamado Bidirectional Encoder Representations from

Transformers (BERT) basado en la arquitectura de transformadores y preentrenamiento en un extenso conjunto de datos. El descubrimiento clave de BERT radica en la comprensión contextual bidireccional a diferencia de los modelos anteriores que procesan texto unidireccionalmente.

BERT considera tanto el contexto izquierdo como el derecho de cada palabra en una oración y por lo tanto recolecta información contextual más significativa. El proceso de preentrenamiento permite el aprendizaje de representaciones de palabras contextualizadas. Esta investigación destaca la eficacia del preentrenamiento en diversas tareas mediante el ajuste fino del modelo BERT preentrenado en tareas específicas posteriores.

Conjuntamente a lo anterior, existen modelos adicionales construidos sobre el modelo BERT y centrados en el preentrenamiento optimizado. Tal es el caso de Robustly Optimized BERT Pretraining Approach (RoBERTa) [7] cuyo objetivo principal es mejorar la metodología de preentrenamiento de BERT y afrontar las limitaciones. Los investigadores también enfatizan la importancia de entrenar con secuencias más largas y utilizar conjuntos de datos más grandes para el preentrenamiento.

Los mecanismos de atención en las redes de transformadores pueden ser extendidas hacia imágenes estáticas, como evidencian las arquitecturas de modelos de visión preentrenados como ViT [6], BEiT [2] y Swin [8]. Las investigaciones afirman que los transformadores pueden capturar dependencias y relaciones globales dentro de una imagen y por lo tanto son convenientes para el reconocimiento de imágenes a gran escala. Además, los transformadores también pueden ser integrados en el campo de la visión por computadora para modelar datos de video. Javier Selva [15] explora el uso de grandes redes neuronales convolucionales (Convolutional Neural Networks, CNNs) como la base de Vision Transformers (VT); aprovechando los sesgos inductivos y capacidades de reducción de dimensionalidad.

Esta investigación también destaca la explotación explícita de la estructura del video a través de la tokenización. Sin embargo, a pesar de que existen modelos para introducir estrategias específicas en el desarrollo de interacciones espacio-temporales detalladas; aún es necesario la exploración de estrategias para la LSM cuya orientación principal sea el movimiento relativo de las manos mas allá de las características de la imagen. Necati C. [4] presenta una metodología innovadora denominada Sign Language Translation Transformer (SLTT).

Esta metodología incorpora una capa de incrustación espacial y utiliza CNNs para el procesamiento de imágenes. Sin embargo, este enfoque depende de un conjunto de datos completo para lograr un rendimiento robusto y que las CNNs usadas en el modelo puedan extraer información espacial de toda la imagen ocasionando dependencias inherentes en el contexto ambiental. Por lo tanto, hemos considerado las investigaciones de Necati como punto de partida fundamental para nuestra propuesta de extender los articuladores de señas a través de puntos de referencia en un modelo preentrenado que pueda ser ajustado de manera fina.

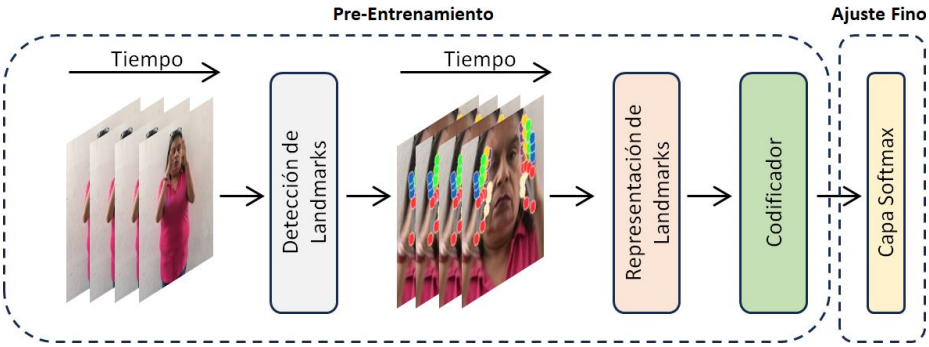


Fig. 1. Estructura general de la estrategia ELT.

3. ELT

El modelo ELT propone una estructura general en donde la representación de señas dinámicas es llevada a cabo dentro de una secuencia de imágenes descritas como una secuencia de puntos de referencia de mano en un espacio vectorial que alimenta a un codificador de las redes de transformadores, como es mostrada en la Figura 1. En conjunto con mecanismos de atención y codificación posicional, posee la capacidad de adquirir los matices cinemáticos que constituyen gestos del léxico de la LSM.

3.1. Puntos de referencia embebidos

Una representación exhaustiva de puntos de referencia en las manos es expuesta de manera profunda en [9]. Este modelo extrae 21 puntos clave de ambas manos. Cada punto clave consta de tres coordenadas espaciales $\{x,\ y,\ z\}$. Las coordenadas $\{x,\ y\}$ se normalizan al rango $\{0{,}0,\ 1{,}0\}$ en relación con la imagen. La coordenada $\{z\}$ es normalizada con la misma magnitud que la coordenada $\{x\}$ para generalizar y detectar correctamente los puntos en una variedad de situaciones y condiciones. Los 21 puntos extraídos de la mano forman un lexema.

Un lexema puede expresarse como un vector continuo en un espacio dimensional en donde cada elemento posee características distintas. Los algoritmos como word2vec emplean un enfoque estructurado para la extracción de características. Nuestro enfoque conlleva el aprovechamiento de los detalles intrincados de gestos de las manos en lugar de la dependencia de características lingüísticas tradicionales. De manera particular, son considerados 21 puntos clave asociados con los movimientos de las manos y, adicionalmente, es construido un nuevo vector, denominado E_{lado} en (1), a través de la concatenación de estos puntos clave. El vector resultante es caracterizado por una dimensionalidad de 63:

$$E_{\text{lado}} = \bigcup_{i=0}^{20} [x_i, y_i, z_i]. \tag{1}$$

La extracción de puntos clave de la segunda mano mejora la integralidad de la representación de características.

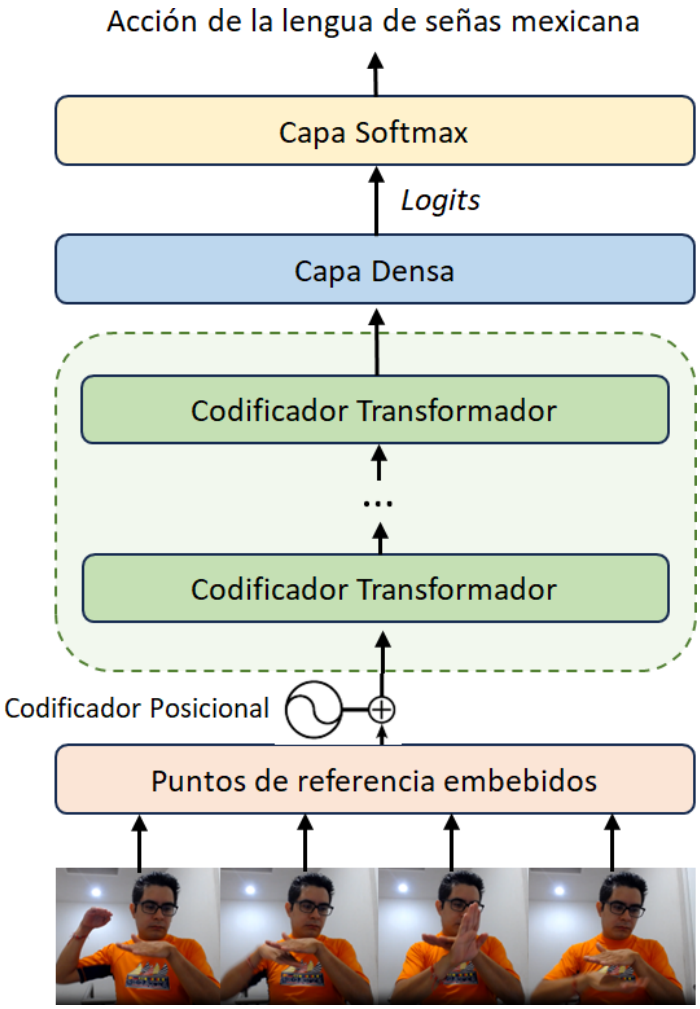


Fig. 2. Arquitectura general del modelo ELT.

Este proceso produce un vector de características dimensionales de 126 y por lo tanto engloba un conjunto rico de atributos espaciales y temporales. La ampliación de información a través de este enfoque de mano izquierda E_{I} y mano derecha E_{D} contribuye en la robustez y la precisión del sistema general de reconocimiento de LSM, como es mostado en (2):

$$E = [E_{\mathrm{I}},\ E_{\mathrm{D}}]. \tag{2}$$

Es importante mencionar que, en el caso de señas llevadas a cabo con una sola mano, el algoritmo rellena el vector E con 63 valores de 10^{-9} cada uno.

3.2. Arquitectura del modelo

El modelo ETL, inspirado en el trabajo fundamental de Vaswani [17] sobre las redes de transformadores, sigue una estructura semejante al modelo original del codificador. De manera notable, la capa Input Embedding en [17] es reemplazada con la capa de Puntos de referencia embebidos, como es mostrado en la Figura 2, ocasionando un cambio hacia las entradas de puntos de referencia en imágenes. En la configuración propuesta, especificamos parámetros para mejorar las capacidades del modelo.

Esto incluye un codificador de 12 capas ($L = 12$), 14 cabezas de autoatención ($A = 14$) para mecanismos de atención robustos, y una dimensión de capa oculta de 126 ($H = 126$). La elección de la dimensión L está alineada deliberadamente con el modelo ampliamente utilizado en [5] y por lo tanto promueve la consistencia y la comparabilidad en las métricas de rendimiento. Debido a que cada vector E contiene la representación de las manos en un momento específico de una secuencia S de imágenes, la secuencia completa para un gesto de lenguaje de señas debe estar compuesta por n vectores E_n, como es mostrado en (3), en donde el número máximo de vectores está establecido por el valor de longitud de secuencia:

$$S = [E_0,\ E_1,\ E_2,\ \ \ldots,\ E_n]. \tag{3}$$

3.3. Etapa de preentrenamiento

Durante la etapa de preentrenamiento, el modelo utiliza 60 vectores para representar el movimiento de una secuencia de señas. Habitualmente, una seña tiene una duración mínima y máxima entre 1 y 2 segundos. En el caso de suponer una duración promedio de 30 cuadros por segundo para un video, el proceso completo consume un total de 60 cuadros y por lo tanto corresponde con la secuencia propuesta. En una situación donde la duración de la seña es inferior a dos segundos, la secuencia es completada mediante un vector de relleno especial [PAD] para asegurar una dimensionalidad consistente en la secuencia S. El valor para cada elemento en el vector [PAD] es 10^{-9} [17].

El entrenamiento del modelo ELT es llevado a cabo con la estrategia propuesta por el modelo BERT, la cual está relacionada con tareas no supervisadas. De manera similar, un vector especial [MASK] es utilizado para enmascarar de forma aleatoria un vector E para una secuencia de entrada dada S. Por lo tanto, el proceso de entrenamiento implica predecir el vector E enmascarado usando el contexto de la secuencia. El valor propuesto para cada elemento en el vector [MASK] es 0. Es importante mencionar que, el modelo ELT mantiene la misma dimensión de entrada a diferencia de BERT que conecta la salida a una capa lineal de tamaño de vocabulario para generar logits.

3.4. Ajuste fino para ELT

El proceso de ajuste fino para el modelo ELT es llevado a cabo de manera fluida mediante la incorporación de un clasificador que consta de una capa Softmax, capas lineales adicionales y una función de activación Tanh. Esta estrategia aumenta la adaptabilidad del modelo hacia tareas específicas. La capa lineal, con una dimensionalidad de 2048, está alineada con las recomendaciones presentadas por las redes de avance de posición en el contexto de los transformadores.

Tabla 1. Distribución de los elementos gramaticales en MX-ITESO-100.

Elementos gramaticales	Cantidad
Verbos	30
Adjectivos	29
Sustantivos	25
Adverbios	6
Pronombres	5
Frases	4
Conjunciones	1

Con respecto a la alimentación del clasificador, solo es considerado el primer vector E_0 de la secuencia de salida S. Es importante establecer que, los hiper parámetros para el ajuste fino están estrechamente alineados con los usados en la etapa de preentrenamiento y por lo tanto mantienen la consistencia en la configuración general del modelo. El único ajuste cuantitativo tiene lugar en la tasa de aprendizaje del optimizador Adam [3], de un valor de tasa de 10^{-2} hacia un valor de 10^{-4}, debido a una actualización en el proceso de convergencia. Finalmente, el costo de entrenamiento durante el ajuste fino es significativamente menor en comparación con el de la etapa de preentrenamiento. Esto es debido en parte a que el modelo ha aprendido, a lo largo del proceso de preentrenamiento, la posición relativa de los puntos de referencia en la imagen con respecto a una secuencia.

3.5. Conjunto de datos

El conjunto de datos MX-ITESO-100 [10] consta de 5000 instancias de video para representar los 100 elementos más importantes del léxico mexicano. A diferencia de compilaciones similares, MX-ITESO-100 prioriza un léxico mexicano amplio y heterogéneo, incluyendo elementos gramaticales esenciales. Cada videograbación engloba una noción conceptual transmitida a través de gestos dinámicos; los cuales abarcan dos etapas secuenciales que van desde configuraciones preliminares hasta configuraciones conclusivas. La distribución de los elementos gramaticales son mostrados en la Tabla 1. Con las videograbaciones previamente descritas, los puntos de referencia de las manos de cada fotograma son extraidos usando la biblioteca MediaPipe [9]. El preprocesamiento de los fotogramas conlleva seleccionar puntos clave dentro de una región central; excluyendo todos los puntos que permanecen afuera de la siguiente región normalizada:

$$1 - T \geq \{x, y\} \geq T, \tag{4}$$

donde T representa un parámetro de umbral. De acuerdo a la expresión en (4) $T = 0{,}25$. Además, [1.0] denota la máxima relación ancho/alto del fotograma. Es importante mencionar que, si no se identifican los puntos de referencia en las manos, el proceso continua con el siguiente fotograma.

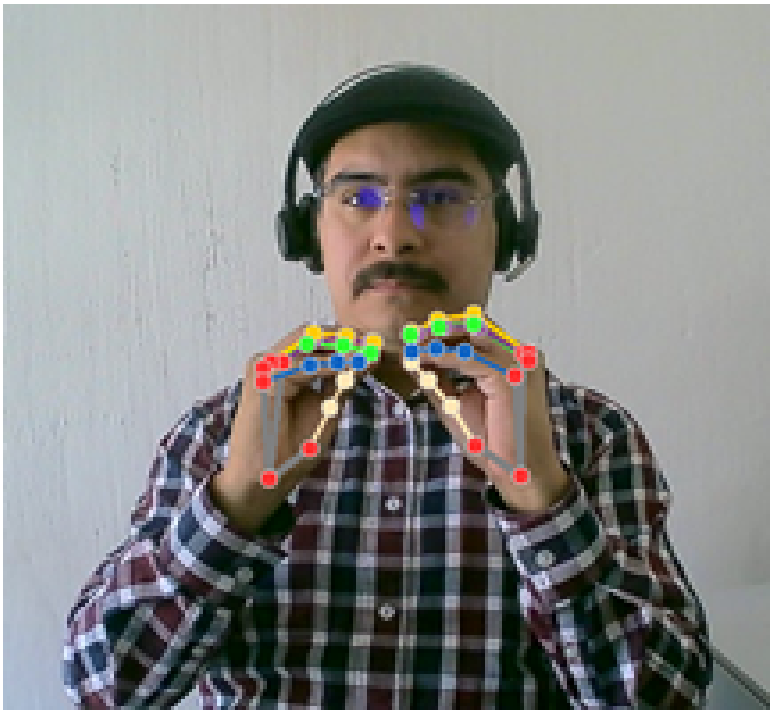


Fig. 3. Reconocimiento de la seña BESAR usando ELT.

4. Experimentación

Durante la etapa experimental, esta investigación usa el conjunto de datos MX-ITESO-100 que consta de 5000 videograbaciones categorizados en 100 elementos gramaticales con 50 videograbaciones para cada elemento gramatical. La arquitectura de modelo consta de 12 capas de codificador, cada una equipada con 14 cabezas de atención. La tasa de abandono (dropout) para cada capa ha sido establecida en 10 % con el propósito de prevenir el sobreajuste. El algoritmo de optimización es Adam con una tasa de aprendizaje de 10^{-2}, $\beta_1 = 0{,}9$ y $\beta_2 = 0{,}98$.

La fase de preentrenamiento incluye 500 épocas mientras que el ajuste fino es llevado a cabo durante 100 épocas. Con respecto a el ajuste fino del clasificador, usamos una función de activación tangente hiperbólica después de una capa lineal con una dimensionalidad de 2048. La función de activación muestra una mejor sensibilidad en valores negativos y por lo tanto el impacto en el sesgo de la red es reducido. Para el optimizador, la tasa de aprendizaje ha sido ajustada en 10^{-4}.

Posteriormente, la secuencia de capas continua con una capa de dropout con valor de 10 %, una capa lineal con 100 neuronas de salida y una capa Softmax para la clasificación final. La Figura 3 muestra la predicción de una seña con el modelo ELT. Los procedimientos de entrenamiento son ejecutados en un sistema Intel XEON habilitado con 128 núcleos, 1TB de memoria RAM y un motor de aceleración Tile Matrix Multiply (TMUL) [1].

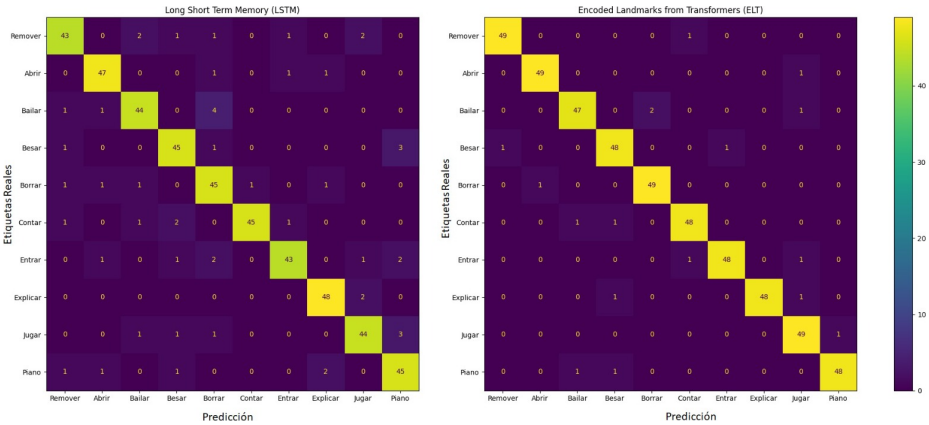


Fig. 4. Matrices de confusión para ETL y LSTM.

La duración del entrenamiento es aproximadamente de 1.2 minutos por época durante la fase de preentrenamiento y en total alrededor de 10 horas. El ajuste fino para 10 gestos exhibe una duración notablemente reducida de 0.13 segundos por época con un tiempo de entrenamiento total aproximado de 2.5 minutos. Adicionalmente, un modelo LSTM bidireccional ha sido entrenado para realizar una comparación de rendimiento. Al igual que el modelo ELT, el modelo LSTM es alimentado con los puntos de referencia extraídos del conjunto de datos MX-ITESO-100. Además, capas TimeDistributed son añadidas a la red LSTM para procesar cada muestra de manera individual. La duración del entrenamiento es aproximadamente de 2.8 minutos por época durante 300 épocas.

Para este caso no hubo un ajuste fino. Por lo tanto, el tiempo de entrenamiento ha sido de 14 horas aproximadamente. La Figura 4 muestra las matrices de confusión producidas por los modelos ETL y LSTM. Para ambos casos, son mostrados únicamente 10 señas relacionadas en su mayoría con verbos. En el caso del modelo clásico LSTM bidireccional, ha sido entrenado únicamente para estas 10 señas ya que no es posible llevar a cabo un ajuste fino. Con relación a la precisión de clasificación, el modelo ELT alcanza un valor de 0.9816 mientras que la precisión para el modelo bidireccional LSTM es de 0.9482. Finalmente, de acuerdo con los resultados experimentales, es posible establecer que el modelo ELT es 28 % mas rápido que el sistema tradicional LSTM durante el proceso de entrenamiento.

5. Conclusiones

En esta investigación, hemos introducido el modelo ELT basado en puntos de referencia de manos usando mecanismos de atención integrados en redes de transformadores. Esta estrategia establece un fundamento sólido para la futura comprensión de la lengua de señas en cualquier contexto lingüístico. El modelo preentrenado puede ser utilizado para la clasificación de señas dinámicas con un costo computacional relativamente modesto y una precisión significativamente elevada por encima del 97 % para un léxico compuesto de 100 elementos gramaticales. Los

resultados experimentales establecen un precedente en la lengua de señas mexicana para contribuir a la mejora en la comunicación de millones de personas sordas y en la integración hacia las actividades sociales rutinarias. Finalmente, el trabajo futuro está relacionado con la representación de puntos de referencia para las expresiones del rostro y la postura del cuerpo. Además, es necesario generar un modelo preentrenado con un léxico más robusto y regionalismos típicos.

Referencias

1. Bal, S., Mummidi, C.S., Da-Cruz-Ferreira, V., Srinivasan, S., Kundu, S.: A novel fault-tolerant architecture for tiled matrix multiplication pp. 1–6 (2023)
2. Bao, H., Dong, L., Piao, S., Wei, F.: BEit: BERT pre-training of image transformers. In: International Conference on Learning Representations. pp. 1–18 (2022)
3. Bernico, M.: Deep learning quick reference: Useful hacks for training and optimizing deep neural networks with TensorFlow and Keras. Packt Publishing (2018)
4. Camgoz, N.C., Koller, O., Hadfield, S., Bowden, R.: Sign language transformers: Joint end-to-end sign language recognition and translation pp. 10023–10033 (2020)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North. vol. 1, pp. 4171–4186 (2019)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16×16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations. pp. 1–21 (2021)
7. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: A robustly optimized BERT pretraining approach. In: International Conference on Learning Representations. pp. 1–15 (2019)
8. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: IEEE/CVF International Conference on Computer Vision. pp. 9992–10002 (2021)
9. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.L., Yong, M.G., Lee, J., Chang, W.T., Hua, W., Georg, M., Grundmann, M.: MediaPipe: A framework for perceiving and processing reality. In: 3rd Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition. pp. 1–4 (2019)
10. Martínez-Sánchez, V., Villalón-Turrubiates, I., Cervantes-Álvarez, F., Hernández-Mejía, C.: Exploring a novel mexican sign language lexicon video dataset. Multimodal Technologies and Interaction 7(8), 83 (2023)
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: International Conference on Learning Representations. pp. 1–12 (2013)
12. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. vol. 1, pp. 2227–2237. Association for Computational Linguistics (2018)
13. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018), paperswithcode.com/paper/improving-language-understanding-by
14. Rothman, D., Gulli, A.: Transformers for natural language processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more. Packt Publishing (2022)

15. Selva, J., Johansen, A.S., Escalera, S., Nasrollahi, K., Moeslund, T.B., Clapés, A.: Video transformers: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 45(11), 12922–12943 (2023)
16. Tunstall, L., von-Werra, L., Wolf, T.: Natural language processing with transformers. O'Reilly Media (2022)
17. Wu, Y.: Attention is all you need for boosting graph convolutional neural network (2024)

ISSN 1870-4069

Detección y clasificación de actividades de la vida diaria y de caídas en personas mediante lógica difusa

Edmundo Bonilla-Huerta, Eduardo Martínez-Juárez,
Roberto Morales-Caporal, Eduardo Vázquez-Urbina

Tecnológico Nacional de México,
Campus Apizaco,
México

{edmundo.bh, m23370044, roberto.mc,
m23370050}@apizaco.tecnm.mx

Resumen. En este artículo se analizan los movimientos de personas jóvenes y de la tercera edad utilizando datos recolectados a partir de un acelerómetro y un giroscopio. Se propone, en este estudio, un enfoque basado en la lógica difusa, para la clasificación de movimientos normales y de caídas. Los resultados obtenidos muestran que la fusión de los datos de un acelerómetro y un giroscopio pueden ser integrados en un sistema difuso para clasificar caídas y movimientos de la vida diaria con un $97{,}4\,\%$ de precisión.

Palabras clave: Tercera edad, acelerómetro, giroscopio, caídas, lógica difusa.

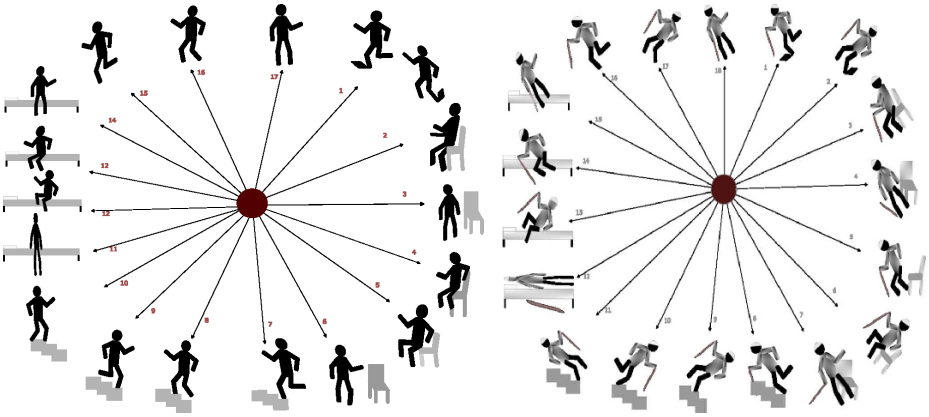
Detection and Classification of Daily Living Activities and Falls in People, Using Fuzzy Logic

Abstract. In this paper, the movements of young and elderly people are analyzed using data collected from an accelerometer and a gyroscope. In this study, a fuzzy logic-based approach is proposed for the classification of normal movements and falls. The results obtained show that the fusion of data from an accelerometer and a gyroscope can be integrated into a fuzzy system to classify falls and daily life movements with a 97.4% accuracy.

Keywords: Elderly people, accelerometer, gyroscope, falls, fuzzy logic.

1. Introducción

Las caídas son comunes entre las personas mayores y representan un desafío significativo para los sistemas de salud pública, generando costos en hospitalización, rehabilitación, atención domiciliaria, y más. Un sistema basado en sensores inerciales de bajo costo podría ser una herramienta eficaz para detectar caídas en esta población. La investigación sobre la actividad humana, en particular, las caídas y la utilización de dispositivos portátiles ha experimentado un desarrollo interesante en



(a) Actividades normales de jóvenes. (b) Caídas de Adultos mayores.

Fig. 1. Actividades de la vida diaria.

los últimos años. Sin embargo, hay pocos conjuntos de datos de libre acceso, todos registrados con teléfonos inteligentes, que sean eficientes, debido a la falta de datos de la población objetivo tales como la edad, su condición de salud, altura, peso entre otros.

Los movimientos de las actividades que realiza un adulto mayor en la vida diaria, generalmente son lentos, y si está asociados a un tipo de distrofia muscular o debilitamiento en las articulaciones, estos pueden llegar a ser muy lentos. Esto supone un desafío para detectar y clasificar estos movimientos, ya que el cambio de aceleración en una caminata no será muy notorio utilizando un acelerómetro o un giroscopio. Un ligero cambio de aceleración podría verse reflejado en un pico en el eje Y de un acelerómetro, sobre todo si la persona va a sentarse, va caminando, o incluso si tropezó con un objeto en casa o en la calle, o simplemente resbaló y la caída es inevitable.

Los acelerómetros triaxiales recolectan la aceleración lineal de tres ejes principales: la aceleración hacia adelante (eje y), la aceleración horizontal (eje x) y la aceleración vertical (eje z). Los acelerómetros proporcionan información de la aceleración lineal de una persona, cuando esta realiza diferentes tipos de movimientos. Por su parte un giroscopio tiene como propósito registrar datos de la velocidad angular, sobre todo si la persona realiza un cambio de rotación en un eje, porque esto indicaría que la persona se ha tropezado o resbalado, y ha sufrido una caída.

Si un girosocopio y un acelerómetro se acondicionan como un dispositivo externo en el cuerpo de una persona, este realiza mediciones de la velocidad angular en estrecha relación con el cuerpo. Las ventajas de utilizar un acelómetro y un giroscopio es debido a su bajo costo, la miniturización de estos dispositivos, su bajo consumo de energía, su integración con otros dispositivos (una cámara, un sensor auditivo, una alarma vía bluetooth o la comunicación con otros dispositivos en la nube), y por su viabilidad de implementarlos en una tarjeta de control para monitorear actividades de la vida diaria (AVD).

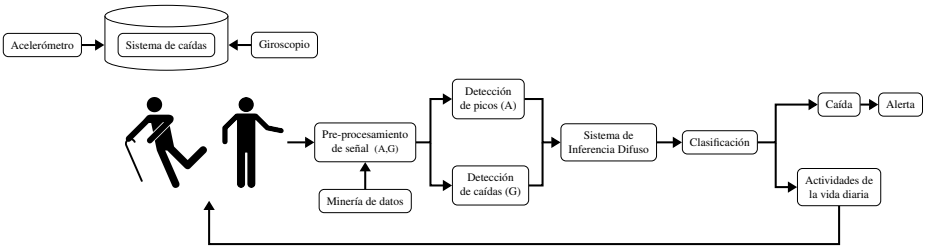


Fig. 2. Sistema propuesto para la detección de caídas y actividades de la vida diaria.

2. Estado del arte

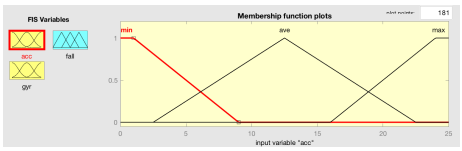
En este artículo [14], se presenta un sistema para detectar caídas, el cual es accesible y de bajo costo que utiliza sensores inerciales. Los sensores inerciales recopilan datos para identificar y detectar 4.444 condiciones de caída diferentes. Se probaron diferentes clasificadores de aprendizaje automático en el conjunto de datos de entrenamiento y se utilizó el mejor clasificador para entrenar los datos del sensor. Posteriormente, el modelo se comparó con datos de sensores desconocidos (recopilados de conjuntos de datos disponibles) para inferir en qué estado se encontraba la persona.

Los autores argumentan que, este sistema podría usarse potencialmente para la detección de caídas. En este artículo [11] se propone un procedimiento de detección de caídas, la cual consta de una unidad inercial que incorpora un acelerómetro triaxial, un giroscopio y un magnetómetro con combinación de información para realizar cálculos para conocer la ubicación de una caída. El algoritmo se ha probado en diferentes escenarios: los voluntarios realizaron recrearon caídas y ejercicios de la vida cotidiana. Al colocar el sensor portátil en el abdomen del sujeto, el dispositivo puede detectar caídas mejor que otras propuestas en la literatura.

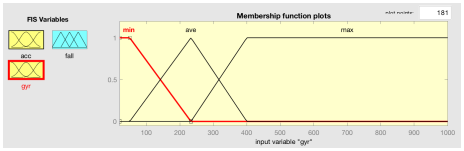
Los resultados obtenidos son muy aceptables en las medidas de exactitud, sensitividad y especificidad. En este artículo [16] se reporta un repositorio de caídas y ejercicios de la vida diaria (EVD) obtenidos con un dispositivo de desarrollo propio compuesto por un acelerómetro y un giroscopio. Esta base se compone de 19 EVD y 15 tipos de caída realizados por 23 adultos jóvenes, 15 tipos de EVD realizados por 14 personas mayores de 62 años, y datos de una persona de 60 años que realizó todas las EVD y caídas. Estos ejercicios se eligieron en base a un estudio y un examen escrito.

Se probó el conjunto de datos con un clasificador basado en umbrales, logrando hasta un 96 % de precisión. Un enfoque basado en lógica difusa, se reporta en [3], en donde se calcula el descubrimiento de caídas mediante la localización basada en un acelerómetro y un sensor de sonido para detectar una posible una caída.

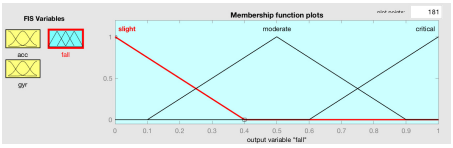
Sin embargo, se ha demostrado que utilizar solo el acelerómetro no es suficiente para identificar con precisión una caída; ya que el acelerómetro también confunde algunos ejercicios de movimiento diarios, y los clasifica como caídas importantes. De esta manera, se crea un cálculo de ubicación de caída basado en lógica difusa para clasificar las señales del acelerómetro y el sensor de sonido, y de esta forma inferir si el evento se trata de un caída válida o no.



(a) Entrada: Acelerómetro.



(b) Entrada: Giroscopio.



(c) Salida: Caída.

Fig. 3. Fuzzificación de entradas y salidas del modelo propuesto.

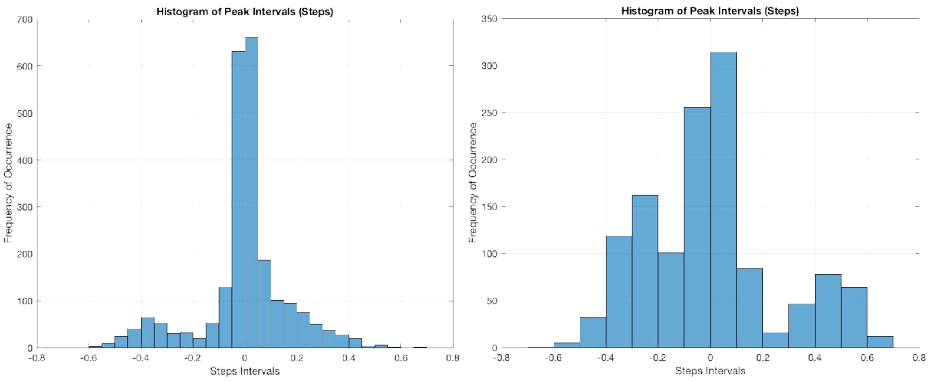
En esta investigación [21] se reporta que la detección de caídas previas al impacto mediante sensores inerciales portátiles ha aumentado debido a su potencial para desarrollar sistemas de prevención de lesiones relacionadas con caídas. Sin embargo, evaluar diferentes algoritmos es un desafío, ya que los investigadores rara vez comparten sus conjuntos de datos. Para abordar esto, los autores han desarrollado y publicado un conjunto de datos de movimiento a gran escala llamado ”KFall”, con etiquetas temporales para el tiempo de caída. Estos recursos pueden ayudar en el desarrollo de metodologías apoyadas con nueva tecnología para la detección de caídas y la prevención de lesiones en las personas mayores.

En este artículo [19], se realiza la comparación de tres tipos de fusión de datos recolectados de un acelerómetro y un giroscopio, para analizar la actividad humana. Este análisis se realizó en cuatro bases de datos públicas utilizando cuatro clasificadores de aprendizaje automático para validar los resultados. Los resultados reportados indican que su modelo de fusión supera algunas otros algoritmos propuestos en la literatura; sin embargo, la carga computacional requerida para el entrenamiento y la clasificación fue alta. Los resultados de esta propuesta sirven como base comparativa para otras técnicas de fusión de datos, aplicadas para el reconomiento de actividades humanas.

En los artículos reportados en [8, 7, 10, 9, 5, 2, 12] se proporciona un análisis detallado de los sistemas de detección de caídas con énfasis en técnicas basadas en fusión multisensor. En su mayoría, utilizan la fusión de los datos de acelerometros, giroscopios y magnetómetros. Algunos otros los fusionan con otros sensores como: sensores infrarrojos, kinect, cámaras, electrocardiogramas, radares, e incluso señales GPS obtenidas de los smartphones. Es importante señalar que en esos trabajos de investigación no se reporta la lógica difusa para realizar la fusión de los datos proporcionados por un acelerómetro y un giroscopio, como se propone en este artículo.

3. Base de movimientos

Para la realización de experimentos, se utilizó la base de caídas SysFall [16], la cual consiste de diferentes tipos de caídas, en la figura 1, se muestra esta información. Las actividades a analizar se enlistan como caídas:



(a) Caminata lenta. (b) Caminata rápida.

Fig. 4. Histogramas de caminata para una persona joven.

1. Hacia adelante al caminar provocada por un resbalón.
2. Hacia atrás al caminar provocada por un resbalón.
3. Hacia atrás al caminar provocada por un resbalón.
4. Lateral al caminar provocada por un resbalón.
5. Hacia delante al caminar provocada por un tropiezo.
6. Hacia delante mientras se trota provocada por un resbalón.
7. Vertical al caminar provocada por desmayo.
8. Caída al caminar, provocada por desmayo.
9. Hacia adelante al intentar levantarse.
10. Lateral al intentar levantarse.
11. Hacia adelante al intentar sentarse.
12. Hacia atrás al intentar sentarse.
13. Lateral al intentar sentarse.
14. Hacia delante estando sentado, provocada por desmayo o quedarse dormido.
15. Hacia atrás estando sentado, provocada por desmayo o quedarse dormido.
16. Lateral estando sentado, provocada por desmayo o quedarse dormido.

4. Modelo propuesto

El modelo que se propone utiliza una base de movimientos disponible en internet y ampliamente referenciada en la literatura. En la figura 2, se muestra el modelo propuesto.

4.1. Magnitud de la señal del acelerómetro y giroscopio

La magnitud, de los datos recolectados por el acelerómetro en cada eje $(x,\ y,\ z)$. Estos ejes se representan comúnmente como vectores de aceleración en estas 3 direcciones: Acel(x), Acel(y) y Acel(z). El cálculo de la aceleración total se obtiene mediante la siguiente ecuación:

$$M(A) = \sqrt{\text{Acel}(x)^2 + \text{Acel}(y)^2 + \text{Acel}(z)^2}. \tag{1}$$

La velocidad angular del giroscopio se realiza en los ejes : roll, pitch y yaw; estos se representan como vectores $\text{Giros}(x)$, Giros(y) y Giros(z) respectivamente. Para el giroscopio se aplica la misma fórmula para calcular la magnitud total:

$$M(G) = \sqrt{\text{Giros}(x)^2 + \text{Giros}(y)^2 + \text{Giros}(z)^2}. \tag{2}$$

4.2. Reducción de la gravedad

Para reducir los efectos de la gravedad en Las señales del acelerómeto, el método utilizado fue el de eliminar el valor medio de su señal. De esta forma se remueven las frecuencias muy bajas y muy altas producidas por los efectos del movimiento:

$$G = M(G) - \overline{M(G)}. \tag{3}$$

4.3. Normalización de la señal

Normalización de los datos en el intervalo $[0, 1]$ para el acelerómetro y el giroscopio utilizando la función max-min:

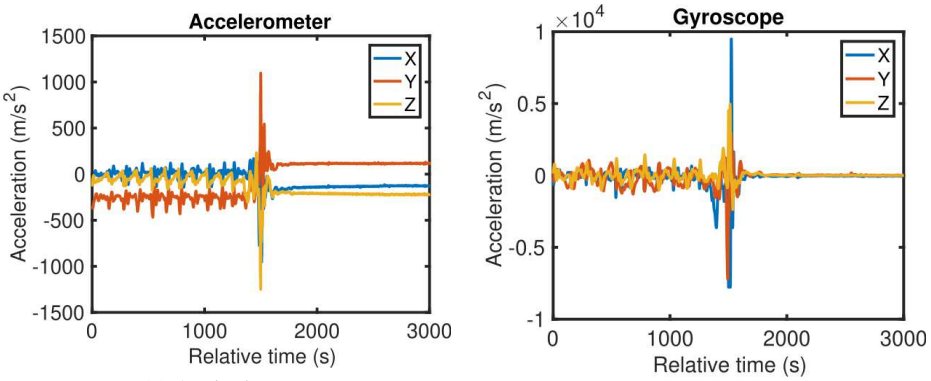
$$NG = \frac{G - \text{mín}\, G}{\text{máx}\, G - \text{mín}\, G}. \tag{4}$$

4.4. Picos de la señal

Se aplica una función para encontrar los picos de la señal de entrada normalizada de la magnitud del acelerómetro y del giroscopio, la cual devuelve un vector que contiene los máximos locales (picos). Un pico local se genera entre el mayor dato de la señal de dos muestras adyacentes. Los picos se generan en el orden en que aparecen. El análisis para descubrir patrones y clasificarlos se encuentra entre la amplitud de estos picos. Los picos detectados indican actividades estacionarias, muy importante si se trata de detectar caídas. Si la distancia entre picos se presentan en intervalos regulares, significa que la marcha de la persona es considerada normal, de lo contrario, hay una posible anomalía que hay que analizar.

4.5. Sistema de inferencia difuso

Un sistema de inferencia difusa (FIS) se puede utilizar como herramienta para identificar y clasificar patrones de los movimiento de personas. Debido a su flexibilidad, facilidad de comprensión y tolerancia a datos imprecisos, en este artículo se emplea un FIS para clasificar caídas y actividades de la vida diaria.



(a) Acelerómetro. (b)Giroscopio.

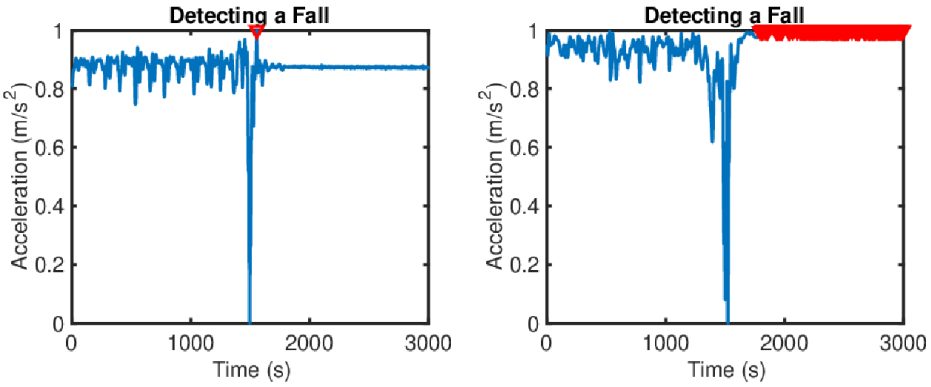
Fig. 5. Velocidades de aceleración en la caída de una persona adulta.

Fuzzificación. Para aplicar el FIS, se definieron 2 entradas y 1 salida (Ver Figuras 3a, 3b, y 3c). la primera entrada fuzzifica los picos obtenidos del acelerómetro, y la segunda las del giroscopio. La salida es la clasificación de la fusión de estas dos señales. La figura 3, ilustra este proceso.

Base de reglas. La parte principal del FIS es definir el conjunto de reglas difusas para identificar los movimientos de la vida diaria de los de una caída. El total de reglas SI-ENTONCES se muestan a continuación:

1. Si Picos-acelerómetro ES Minimo Y Picos-Giroscopio ES Minimo ENTONCES Posible-ADL.
2. SI Picos-acelerometro ES Minimo Y Picos-Giroscopio ES Medio ENTONCES Caída esta en curso.
3. SI Picos-acelerometro ES Minimo Y Picos-Giroscopio ES Máximo ENTONCES Caída detectada.
4. SI Picos-acelerómetro ES Medio Y Picos-Giroscopio ES Mínimo ENTONCES ADL.
5. SI Picos-acelerometro ES Medio Y Picos-Giroscopio ES Medio ENTONCES Posible-ADL.
6. SI Picos-acelerometro ES Medio Y Picos-Giroscopio ES Máximo ENTONCES Caída esta en curso.
7. SI Picos-acelerómetro ES Alto Y Picos-Giroscopio ES Mínimo ENTONCES posible-ADL.
8. SI Picos-acelerometro ES Alto Y Picos-Giroscopio ES Medio ENTONCES ADL.
9. SI Picos-acelerometro ES Alto Y Picos-Giroscopio ES Máximo ENTONCES ADL.

Defuzzificación. Para el proceso de transformación de los activaciones de las reglas difusas, se utiliza el método del centroide o centro del área para conocer el valor de salida. Este valor es la base para realizar el proceso de clasificación.



(a) Acelerómetro. (b) Giroscopio.

Fig. 6. Detección de caída de una persona adulta utilizando los dos sensores.

5. Análisis de resultados del modelo propuesto

En las figuras 4a y 4b, se muestran en forma de histograma, la distribución de los intervalos de picos de una persona joven caminando lento y rápido respectivamente. La distribución de la 4a, indica que el intervalo promedio de picos es de 8.4621. En contraste, en la figura 4b, el intervalo promedio es 15.4868. Estos valores indican los pasos promedio que la persona da en un intervalo de tiempo. En la Figura 5a y 5b, se muestran los datos recolectados del acelerómetro y del giroscopio cuando una persona ha sufrido una caída. El acelerómetro registrado un cambio en los ejes x, y, z; mientras que el giroscopio el eje que ha registrado esta caída es el eje y.

Estos datos son pre-procesados, es decir se obtiene su magnitud, se reduce la gravedad de ellos y se normalizan. Finalmente se encuentran los picos de cada sensor, para saber si se trata de una actividad de la vida diaria o de una caída. Como se ilustra en las Figuras 6a y 6b, los dos sensores complementan la información cuando se registra el evento de una caída de una persona adulta. En la Figura 6a se detecta el pico en el tiempo donde hay una pérdida de equilibrio. Mientras que en la Figura 6b se observa que efectivamente la persona ha caído, y el giroscopio detecta pequeños cambios en la velocidad angular, lo que indica que permanece en el suelo después de la caída. En estos momentos es cuando se activa el sistema de inferencia difuso para clasificar estos datos recolectados del acelerómetro y giroscopio respectivamente.

5.1. Discusión

La evaluación de nuestro modelo fue realizada mediante métricas de sensitividad y especificidad:

$$\text{Sensitividad} = \frac{\text{VP}}{\text{VP} + \text{FN}}, \tag{5}$$

$$\text{Especificidad} = \frac{\text{VP}}{\text{VP} + \text{FP}}, \tag{6}$$

Tabla 1. Comparación de métricas de rendimiento (sensibilidad, especificidad, precisión y F_1 score) entre nuestro modelo y otros estudios previos en detección de caídas.

Modelo	Sensitividad	Especificidad	Precision	F_1 Score
ML algorithms [15]	–	–	–	95.91 %
Optimization [6]	96.30 %	96.20 %	–	–
Threshold [13]	–	–	–	93.3 %
SVM,ANN [4]	90.57 %	96.91 %	–	–
Transformed-based [20]	–	–	–	96.00 %
LSTM-based [20]	–	–	–	97.00 %
Threshold [17]	90.00 %	85.00 %	87.50 %	–
Vision [18]	90.33 %	89.66 %	89.73 %	90.02 %
Clustering-EGG [1]	–	–	–	97.1 %
Nuestro modelo	94.94 %	**100 %**	96.92 %	**97.40 %**

$$\text{Precision} = \frac{\text{VP} + \text{VN}}{\text{VP} + \text{VN} + \text{FP} + \text{FN}}, \tag{7}$$

$$\text{F\_1 score} = \frac{2 \times \text{Precision} \times \text{Sensitividad}}{\text{Precision} + \text{Sensitividad}}, \tag{8}$$

donde VP, VN, FP y FN, corresponden a valores Verdaderos Positivos, Verdaderos Negativos, Falsos Positivos y Falsos Negativos, respectivamente; los cuales son obtenidos de la matriz de confusión, al clasificar actividades de la vida diaria y caídas. VP, indica una caída detectada correctamente. FP, indica una caída mientras se realizaba una actividad de la vida diaria (ADL); mientra que VN indica que no es una caída sino una ADL, y finalmente FN, detecta incorrectamente una caída cuando se trata de una ADL. En la tabla 1, se muestran los resultado de nuestro modelo; así como una comparación con otras metodologías similares reportadas en la literatura; las cuales utilizan como base las lecturas de un acelerómetro y un giroscopio.

La comparación se realizó solo en aquellas metodologías que utilizan las mismas métricas de evaluación. En algunos casos solo reportan alguna de ellas. Las que no se reportan aparecen en la tabla con un guión. Como se puede observar nuestro modelo logra un $100\,\%$ de especificidad para detectar caídas. Se obtiene un F_1 score del $97{,}40\,\%$. Estos resultados son ligeramentamente superiores a otros métodos como los que incluyen umbrales, algoritmos de aprendizaje máquina, redes neuronales y técnicas de agrupamiento entre otros.

6. Conclusiones y trabajos futuros

Nuestro modelo propuesto muestra un desempeño muy bueno, con respecto a la utilización de modelos muy sofisticados como las Máquinas de Soporte Vectorial (SVM), las cuales son generalmente muy difíciles de parametrizar.

Incluso las Redes Neuronales Artificiales, las cuales por su naturaleza estocástica y configuración de capas intermedias, dependen de una tasa de aprendizaje, el cual si el aprendizaje es lento pueden converger en un número de épocas alto. Si el aprendizaje es rápido quedan atrapadas en un mínimo/máximo local y no un mínimo/máximo global. En el futuro se va a diseñar una base de creación propia de movimientos de la vida diaria y caídas. Se tienen contemplados dos escenarios: Actividades de la vida diaria en interiores y exteriores. Las caídas se van a recrear utilizando personas jóvenes que practican gimnasia. Para ello se va a diseñar el aparato que fusione las señales de un acelerómetro, un giroscopio y una magnétometro y enviar señales a través de un celular o un sensor sonoro para activar alertas.

Referencias

1. Al-Dujaili, M. J., Dhaam, H. Z., Mezeel, M. T.: An intelligent fall detection algorithm for elderly monitoring in the internet of things platform. Multimedia Tools and Applications, vol. 83, no. 2, pp. 5683–5695 (2023) doi: 10.1007/s11042-023-15820-0
2. Dentamaro, V., Gattulli, V., Impedovo, D., Manca, F.: Human activity recognition with smartphone-integrated sensors: a survey. Expert Systems with Applications, vol. 246, pp. 123143 (2024) doi: 10.1016/j.eswa.2024.123143
3. Er, P. V., Tan, K. K.: Non-intrusive fall detection monitoring for the elderly based on fuzzy logic. Measurement, vol. 124, pp. 91–102 (2018) doi: 10.1016/j.measurement.2018.04.009
4. Fula, V., Moreno, P.: Wrist-based fall detection: Towards generalization across datasets. Sensors, vol. 24, no. 5, pp. 1679 (2024) doi: 10.3390/s24051679
5. Gharghan, S. K., Hashim, H. A.: A comprehensive review of elderly fall detection using wireless communication and artificial intelligence techniques. Measurement, vol. 226, pp. 114186 (2024) doi: 10.1016/j.measurement.2024.114186
6. Huynh, Q. T., Nguyen, U. D., Irazabal, L. B., Ghassemian, N., Tran, B. Q.: Optimization of an accelerometer and gyroscope-based fall detection algorithm. Journal of Sensors, vol. 2015, pp. 1–8 (2015) doi: 10.1155/2015/452078
7. Khan, S. S., Hoey, J.: Review of fall detection techniques: A data availability perspective. Medical Engineering and Physics, vol. 39, pp. 12–22 (2017) doi: 10.1016/j.medengphy.2016.10.014
8. Koshmak, G., Loutfi, A., Linden, M.: Challenges and issues in multisensor fusion approach for fall detection: Review paper. Journal of Sensors, vol. 2016, pp. 1–12 (2016) doi: 10.1155/2016/6931789
9. Newaz, N. T., Hanada, E.: The methods of fall detection: a literature review. Sensors, vol. 23, no. 11, pp. 5212 (2023) doi: 10.3390/s23115212
10. Nooruddin, S., Islam, M. M., Sharna, F. A., Alhetari, H., Kabir, M. N.: Sensor-based fall detection systems: a review. Journal of Ambient Intelligence and Humanized Computing, vol. 13, no. 5, pp. 2735–2751 (2021) doi: 10.1007/s12652-021-03248-z
11. Pierleoni, P., Belli, A., Palma, L., Pellegrini, M., Pernini, L., Valenti, S.: A high reliability wearable device for elderly fall detection. IEEE Sensors Journal, vol. 15, no. 8, pp. 4544–4553 (2015) doi: 10.1109/jsen.2015.2423562
12. Qu, Z., Huang, T., Ji, Y., Li, Y.: Physics sensor based deep learning fall detection system. arXiv (2024) doi: 10.48550/arXiv.2403.06994
13. Rakhman, A. Z., Nugroho, L. E., Widyawan, Kurnianingsih: Fall detection system using accelerometer and gyroscope based on smartphone. In: Proceedings of the 1st International Conference on Information Technology, Computer, and Electrical Engineering, pp. 99–104 (2014) doi: 10.1109/icitacee.2014.7065722

14. Rodrigues, T. B., Salgado, D. P., Cordeiro, M. C., Osterwald, K. M., Filho, T. F., de-Lucena, V. F., Naves, E. L., Murray, N.: Fall detection system by machine learning framework for public health. Procedia Computer Science, vol. 141, pp. 358–365 (2018) doi: 10.1016/j.procs.2018.10.189
15. Saeed, M. A., Hashem-Almourish, M., Alqady, Y. A., Alsharabi, H., Alkhorasani, H., Alsorori, S., Saeed, A. Y. A.: Predicting fall in elderly people using machine learning. In: Proceedigns of the International Congress of Advanced Technology and Engineering, pp. 1–5 (2021) doi: 10.1109/icoten52080.2021.9493442
16. Sucerquia, A., López, J., Vargas-Bonilla, J.: Sisfall: A fall and movement dataset. Sensors, vol. 17, no. 1, pp. 198 (2017) doi: 10.3390/s17010198
17. Tang, D., Usman, A. B., Abba, A.: Fall detection system with accelerometer and threshold-based algorithm . YHIoT Research Journal, vol. 1, no. 1 (2021)
18. Wang, Y., Deng, T.: Enhancing elderly care: Efficient and reliable real-time fall detection algorithm. Digital Health, vol. 10 (2024) doi: 10.1177/20552076241233690
19. Webber, M., Rojas, R. F.: Human activity recognition with accelerometer and gyroscope: A data fusion approach. IEEE Sensors Journal, vol. 21, no. 15, pp. 16979–16989 (2021) doi: 10.1109/jsen.2021.3079883
20. Yhdego, H., Paolini, C., Audette, M.: Toward real-time, robust wearable sensor fall detection using deep learning methods: a feasibility study. Applied Sciences, vol. 13, no. 8, pp. 4988 (2023) doi: 10.3390/app13084988
21. Yu, X., Jang, J., Xiong, S.: A large-scale open motion dataset (kfall) and benchmark algorithms for detecting pre-impact fall of the elderly using wearable inertial sensors. Frontiers in Aging Neuroscience, vol. 13 (2021) doi: 10.3389/fnagi.2021.692865

ISSN 1870-4069

KMoS-SSA: Gestión sistémica del conocimiento

Jorge Rodas-Osollo[1], Karla Olmos-Sánchez[1], Irina O. Kotlyarova[2], Alicia Jimenez-Galina[1]

[1] Universidad Autónoma de Ciudad Juárez, Instituto de Ingeniería y Tecnología, México

[2] Universidad Estatal de los Urales del Sur, Facultad de Mecánica y Tecnología, Federación Rusa

{jorge.rodas, kolmos, alicia.jimenez}@uacj.mx, kio_ppo@mail.ru

Resumen. En la actualidad, la tecnología desempeña un papel crucial, permeando nuestra vida y marcando una era de transformación digital y cognitiva. Aquellos que adoptan esta transformación ingresan en la llamada Era Cognitiva, donde la Inteligencia Artificial y las Tecnologías de la Información facilitan un soporte sólido a la toma de decisiones. Sin embargo, esta transición genera ansiedad provocando cambios rápidos e inadecuados, impulsados por una simplificación excesiva de dominios complejos para hacerlos más accesibles a la IA y las TI. Este artículo comunica al marco KMoS-SSA como integrador de la Gestión del Conocimiento y del Enfoque Sistémico para desarrollar soluciones basadas en IA, deseables, efectivas y factibles en dominios complejos. Este enfoque transdisciplinar reconoce la importancia del conocimiento tácito de los especialistas del dominio, así como la complejidad de los dominios, ofreciendo una vía conveniente para abordar los desafíos de la transformación digital y cognitiva.

Palabras clave: Transformación digital y cognitiva, gestión del conocimiento, enfoque sistémico, representación del conocimiento, KMoS-SSA.

KMoS-SSA: Systemic Knowledge Management

Abstract. In the current era, technology plays a pivotal role, influencing our lives and marking a period of digital and cognitive transformation. Those who embrace this transformation enter the Cognitive Era, where Artificial Intelligence and Information Technologies provide robust support for decision-making. However, this transition generates anxiety, leading to rapid and inadequate changes driven by an excessive simplification of complex domains to make them more accessible to AI and IT. This paper introduces the KMoS-SSA framework, which integrates Knowledge Management and the Systemic Approach to develop AI-based solutions that are desirable, effective, and feasible in complex domains. This transdisciplinary approach recognises the importance of tacit knowledge from

domain specialists and the complexity of the domains, offering a convenient avenue to address the challenges of digital and cognitive transformation.

Keywords: Digital and cognitive transformation, knowledge management, systemic approach, knowledge representation, KMoS-SSA.

1. Introducción

En el mundo contemporáneo, la tecnología desempeña un papel fundamental, permeando nuestra vida consciente e inconscientemente y marcando una era de transformación digital y cognitiva. En este contexto, las tecnologías digitales utilizan datos, información y conocimiento para impulsar procesos inteligentes y tomar decisiones ágiles, respondiendo en tiempo real o en el menor tiempo posible, a los cambios del entorno. Aquellos que han adoptado la transformación digital se adentran en la denominada Era Cognitiva (EC), donde la Inteligencia Artificial (IA) y las Tecnologías de la Información les permiten gestionar datos, información y conocimiento para tomar decisiones y lograr resultados deseados [29].

Sin embargo, esta transición hacia la EC ha generado ansiedad por aprovechar todas las herramientas disponibles, lo que ha dado lugar a cambios rápidos y transformaciones digitales o cognitivas que pueden resultar inadecuadas. Esto se debe, en gran medida, a una simplificación excesiva de dominios naturalmente complejos para hacerlos más accesibles y garantizar el éxito de las herramientas de IA [15].

No obstante, esta simplificación limita el apoyo a la toma de decisiones estratégicas, por parte de tales herramientas, y a los cambios necesarios en el funcionamiento de los actores del dominio. Por lo tanto, es crucial reconocer las características de los dominios complejos para trabajar adecuadamente con ellas y encontrar soluciones deseables, efectivas y factibles. Una comprensión inadecuada de estos dominios podría conducir al desarrollo de cuasi-soluciones que no abordan eficazmente los problemas y que pueden ocasionar resultados inexactos, pérdida de confianza, costos adicionales, e incluso, riesgos de seguridad.

Este artículo presenta un modelo de proceso que guía la conceptualización, especificación y desarrollo de soluciones inteligentes en dominios complejos, utilizando la Gestión del Conocimiento (GC) y el Pensamiento Sistémico (PS). Este enfoque sistémico no solo considera las diversas perspectivas y soluciones de los actores del dominio, sino también cómo utilizan y comparten la información y el conocimiento. Estas soluciones deben ser efectivas y adaptarse a las necesidades de los usuarios en contextos caracterizados por múltiples elementos interconectados, ambigüedad e incertidumbre, lo que provoca emergencia.

Debido a estas situaciones emergentes, el Conocimiento Tácito (CT) de los especialistas del dominio (ED) adquiere relevancia y requiere una gestión eficaz. A diferencia del enfoque tradicional de la IA, que tiende a simplificar en exceso los dominios complejos, nuestro enfoque aborda estas complejidades de manera integral. La combinación de la GC [21], a través del proceso sistemático KMoS-RE, y PS, a través de métodología de sistemas flexibles (MSF) [4, 18] permite una adaptación continua del conocimiento en respuesta a la evolución y emergencia de las circunstancias.

Este enfoque, especialmente MSF, ha demostrado su eficacia en una variedad de dominios complejos, desde la gestión de residuos médicos hasta la ciberseguridad y la agroindustria del café [30, 33, 8, 1]. El marco integrador KMoS-SSA[26], que se comparte en este documento, se fundamenta en estos principios y metodologías, ofreciendo una vía para abordar los desafíos de la transformación digital y cognitiva de manera efectiva.

2. Antecedentes

2.1. Pensamiento sistémico

El Pensamiento Sistémico (PS) sigue siendo crucial para abordar situaciones complejas al analizar las interacciones entre los elementos de un sistema [7]. Se distingue entre el enfoque del pensamiento sistemático, que se centra en la estructura y el comportamiento de los sistemas, y el pensamiento sistémico, que reconoce la naturaleza interconectada de los elementos de un sistema. Ambos enfoques enfatizan la importancia de adoptar una visión holística y comprender las interrelaciones entre las partes de un sistema y sus principios clave [25] son:

- Enfoque holístico: Aborda las interacciones y relaciones del sistema, reconociendo que el comportamiento sistémico surge de la dinámica entre sus partes.
- Circuitos de retroalimentación: Integra bucles donde la salida de un proceso influye la entrada de otros, amplificando o estabilizando cambios.
- Interconexión: Los componentes están intrínsecamente interconectados, y los cambios en una parte pueden tener efectos en todo el sistema.
- Actuación en contextos complejos: Facilita acciones efectivas en dominios complejos comprendiendo las estructuras y los circuitos de retroalimentación.
- Emergencia: Reconoce propiedades y comportamientos resultantes de las interacciones del sistema.
- Causalidad: Explora relaciones causales complejas y no lineales.
- Límites: Establece los límites del sistema para determinar qué elementos se incluyen.
- Retroalimentación y aprendizaje: Implica mecanismos de retroalimentación para ajustar estrategias en función de los resultados pasados.
- Cambio sistémico: Aborda problemas sistémicos y promueve cambios significativos.

El PS ha demostrado eficacia en diversos ámbitos, desde la gestión hasta la ingeniería y las ciencias sociales [25, 17, 9].

2.2. Metodología de sistemas flexibles

La MSF aborda problemas intrincados en dominios donde no existe una solución única y clara. Se debe considerar como un modelo de aprendizaje que prioriza la comprensión de las percepciones y necesidades de los individuos involucrados,

empleando modelos conceptuales flexibles para explorar diversas perspectivas y formular soluciones deseables, efectivas y factibles. La MSF comprende varios pasos: identificación del problema intrincado, análisis de perspectivas, elaboración de modelos conceptuales flexibles, exploración de soluciones y selección de una solución aceptable. Sus ideas fundamentales [3] son:

– Abordaje de la complejidad del mundo real,
– Reconocimiento de múltiples cosmovisiones,
– Generación de nuevos conocimientos sobre problemas intrincados,
– Selección de soluciones efectivas, factibles y deseables.

La MSF es una herramienta valiosa para enfrentar retos en dominios complejos, que implica la toma de decisiones y la gestión del cambio [25] al comprender la dinámica subyacente y proponer alternativas satisfactorias.

2.3. Gestión del conocimiento

La GC implica obtención del conocimiento explícito a partir del tácito, captura, almacenamiento, distribución y aplicación efectiva del conocimiento dentro de una organización. Se centra en identificar el conocimiento crítico, basado en la experiencia—que suele ser tácito [23], facilitar su intercambio y aprovecharlo para mejorar procesos y decisiones. La GC enfrenta desafíos al gestionar el conocimiento tácito (CT) [19, 17], que abarca el conocimiento intuitivo y experiencial difícil de articular con palabras, pues implica establecer mecanismos para obtenerlo, compartirlo y transferirlo, por ejemplo, entre los miembros de una organización. La integración de conocimientos técnicos es crucial para mejorar la toma de decisiones y fomentar la innovación y el aprendizaje organizativo en ámbitos complejos. En este sentido, años atrás, habiamos integrado a la GC en una estrategia para la ingeniería de requisitos llamada KMoS-RE [20]. Este enfoque permite especificar soluciones en dominios de estructura informal, donde la mayor parte del conocimiento es tácito y derivado de la experiencia. KMoS-RE demostró aplicaciones exitosas en diversos escenarios del mundo real [21, 27].

2.4. Incorporación de MSF en KMoS-RE

Tras explorar y experimentar con herramientas arraigadas en la filosofía del PS, se consideró ventajosa la incorporación de características de MSF en KMoS-RE. Esto implicaba establecer una estructura fundacional que englobara herramientas, métodos y procesos para abordar eficazmente los entresijos de los dominios complejos, tal y como se explica en la sección 3.3. La idea de integrar la tecnología de la información en la GC no es nueva; Nakamori [17] aboga por la colaboración entre ambas disciplinas. Sostiene que, mientras que la GC se ha centrado predominantemente en enfoques sistémicos del conocimiento, lograr la innovación únicamente a través de ésta es todo un reto. Por el contrario, el PS, especializado en abordar problemas intrincados, se enfrenta a retos

formidables cuando se ocupa de dominios complejos y se esfuerza por gestionar piezas de conocimiento afines a las prácticas de la GC. En última instancia, el argumento postula que los esfuerzos de colaboración entre la GC y el PS exigen superar retos sustanciales, entre los que se incluyen éste y otros. Estos retos abarcan la incorporación del conocimiento tradicional al proceso de GC y requieren la participación imperativa de un grupo altamente especializado de profesionales y otros actores involucrados en el proceso emergente. Cabe mencionar que la MSF puede reforzar KMoS-RE al evitar la simplificación excesiva de un dominio complejo y al considerar diversos aspectos como la cultura y el entorno de toma de decisiones. Se ha establecido un marco estratégico llamado **KMoS-SSA**[3] [26] que permite a la GC generar opciones estratégicas mediante un análisis sistémico flexible. Este marco involucra a profesionales especializados y apoya la reflexión continua para lograr soluciones eficientes, deseables y satisfactorias.

3. Dominio complejo

En la EC actual, los modelos que comunican perspectivas del mundo real se integran en ecosistemas cognitivos [29], definiendo dominios donde se toman decisiones y se abordan situaciones problemáticas [27]. Los actores, al enfrentarse a estas soluciones, emplean funciones mentales para tomar decisiones basadas en el CT, como la heurística, las reglas empíricas y las emociones. Sin embargo, con el avance de la tecnología informática, ha habido una simplificación excesiva de los modelos y análisis en diversos ámbitos, excluyendo la incorporación del pensamiento humano, la inteligencia y el conocimiento experiencial en la conceptualización y especificación de soluciones [15].

A modo de ejemplo, consideremos el estrés laboral, el cual constituye un problema grave y recurrente a nivel global. Este dominio puede dar lugar a comportamientos emergentes imprevisibles, como el agotamiento emocional y la despersonalización, que se manifiestan a través de síntomas tales como ansiedad, depresión, trastornos del sueño y problemas cardiovasculares. Asimismo, puede repercutir negativamente en la productividad y la eficiencia organizativa, generando ausentismo, rotación de personal, disminución del compromiso laboral y deterioro del rendimiento y desempeño laborales. No obstante, también puede tener un impacto positivo en la organización, aunque determinar hasta qué punto es apropiado o no, constituye una tarea compleja [22].

El desarrollo de soluciones inteligentes en este ámbito implica la consideración de una diversidad de datos, información y perspectivas o cosmovisiones (CV), dado que el estrés laboral puede ser percibido de manera diferente por los empleados, supervisores y la alta dirección, contribuyendo así a su naturaleza ambigua. Es importante destacar que, por lo general, estas soluciones deben ser llevadas a cabo por un grupo de proveedores especializados en soluciones, quienes suelen ser neófitos en el ámbito del estrés laboral, lo que aumenta la complejidad al requerir técnicas de gestión y representación del conocimiento para llegar a consensos, conceptualizar, diseñar e implementar dichas soluciones.

[3] **KMoS-SSA** del inglés Knowledge Management of Strategic options through Soft Systemic Analysis.

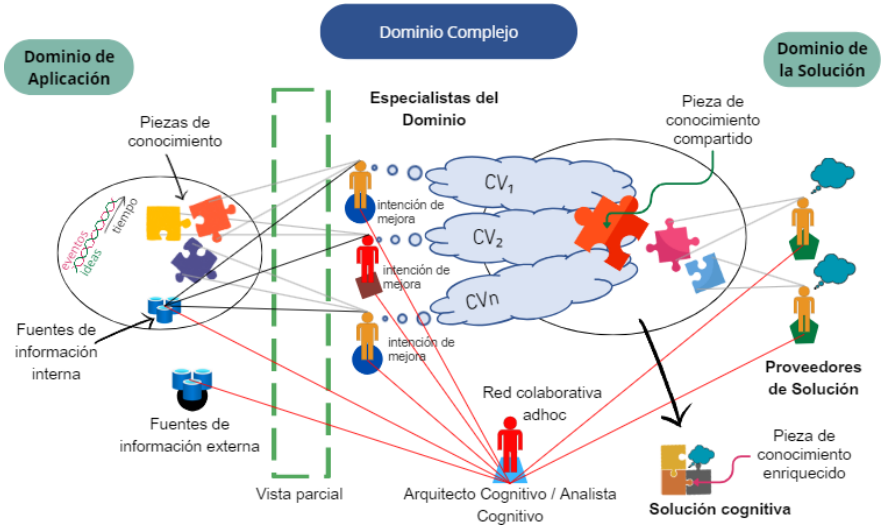


Fig. 1. Vista de características de un DCEI, que incluye un dominio de aplicación y un dominio de solución. Los expertos del dominio (ED) participan activamente, aportando conocimientos parciales según sus roles, que pueden ser varios. Su conocimiento es principalmente tácito, informal y no estructurado. Los conceptos y relaciones en el dominio son ambiguos y derivan de la experiencia de los ED. El DCEI es dinámico y emergente, con un flujo interactivo de eventos, ideas y conocimientos influenciados por fuentes internas y externas. El dominio de solución involucra proveedores, dirigidos por el Analista Cognitivo, que supervisa la colaboración entre los ED. Los proveedores aportan conocimientos para desarrollar soluciones tangibles o intangibles basadas en el conocimiento enriquecido.

3.1. Definiendo al dominio complejo

Un Dominio Complejo de Estructura Informal (DCEI) se caracteriza por la presencia de actores, sus procesos cognitivos, comportamientos e interacciones. Los componentes de este dominio muestran intrincadas interconexiones, abordando diversos niveles de conocimiento y experiencia, con límites difusos. El trabajo colaborativo se desarrolla de forma social, cultural, intuitiva y consensuada. Los actores comparten interconexiones que contienen información explícita y conocimientos tácitos, contribuyendo a comprender la naturaleza del problema o necesidad. Surgen múltiples cosmovisiones del fenómeno, generando alternativas para abordarlo, con o sin soluciones algorítmicas.

Afrontar los retos de un DCEI requiere la pericia de un equipo especializado, ya que los datos, la información y el conocimiento dentro de este dominio son heterogéneos y predominantemente tácitos. Además, el DCEI se caracteriza por altos niveles de ambigüedad e incertidumbre [27]. La comunicación eficaz entre los actores del DCEI es difícil debido a estas características y a la presencia de diferentes dominios de conocimiento en él. Por lo tanto, es esencial un procedimiento sistemático y de pensamiento sistémico para establecer un lenguaje común y garantizar una comunicación de alto nivel, lo que puede requerir una importante inversión de tiempo y recursos.

Los ED tienen cosmovisiones particulares de los dominios y de las situaciones problemáticas, según se pueden apreciar en la representación visual del DCEI, figura 1, como nubes etiquetadas "CV", lo que exige su compromiso para resolver la problemática o atender las necesidades. A pesar de tener piezas comunes de conocimiento es imperativo que cada ED este comprometido a resolver la problemática o atender las necesidades pues por lo general es un conocimiento consensuado el que prosperara dando forma a la alternativa de solución.

3.2. Ciclo de enriquecimiento del conocimiento

El ciclo de enriquecimiento del conocimiento se representa como un proceso cíclico de bucles simétricos que convergen en el centro, formando una estructura horizontal infinita, como se ilustra en la Figura 2. Este ciclo implica dos conjuntos de etapas de límites difusos: las alineadas con el PS y las asociadas con las aplicaciones del mundo real. Las acciones, en cada etapa, giran en torno a la adquisición, ampliación y perfeccionamiento constante del conocimiento del dominio para sustentar la conceptualización y construcción de alternativas para resolución de problemas o atención de necesidades. Aunque los detalles específicos de cada conjunto de etapas pueden variar en función de las características del DCEI analizado, un conjunto fundamental de etapas que contempla el PS incluye típicamente:

– **Elicitación del Conocimiento (ElicCon):** Proceso sistemático de identificación, obtención y organización de información relevante para comprender un dominio.
– **Enriquecimiento del Conocimiento (EnrCon):** Reflexión exhaustiva y validación de las piezas de conocimiento recopiladas, evaluando su pertinencia y conexión con la problemática desde la cosmovisión de cada ED.
– **Generación de Modelos (GenMdl):** Generación de modelos para representar la estructura y dinámica del dominio.
– **Discusión del Modelo (DiscMdl):** Presentación y explicación de modelos para validar su precisión y fomentar la reflexión entre los actores.
– **Validación del Modelo (ValMdl):** Evaluación crítica de la correspondencia del modelo con la realidad y retroalimentación para mejoras.

Este ciclo promueve el aprendizaje continuo y la mejora iterativa del conocimiento del dominio, facilitando la toma de decisiones informadas. Sus acciones suelen llevarse a cabo de manera iterativa y colaborativa, involucrando a diversos actores y utilizando herramientas específicas basadas en los requisitos del análisis sistémico.

3.3. Esquema de trabajo guiado por MSF

Esta subsección presenta una visión general del marco KMoS-SSA quien desarrolla un esquema de trabajo estructurado que incorpora los principios y herramientas de la GC y MSF, entre otros. A continuación se describe la secuencia de acciones como se aprecia en la Figura 3.

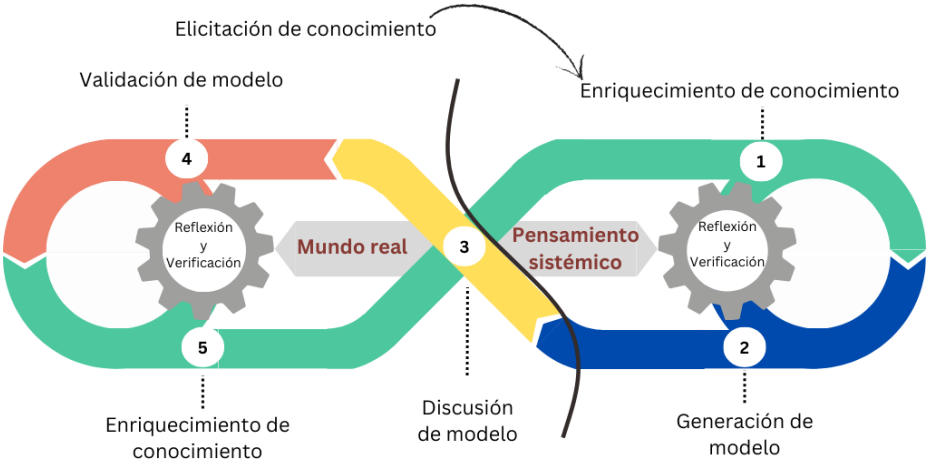


Fig. 2. El ciclo de enriquecimiento de conocimientos se representa visualmente como un proceso cíclico de bucles simétricos que convergen en el centro, formando una estructura horizontal infinita. Este ciclo consta de dos conjuntos de etapas con límites difusos: aquellas alineadas con el pensamiento sistémico y las conectadas con alternativas del mundo real. Dentro del ciclo, las etapas activas se identifican mediante círculos numerados y colores asociados. Es de suma importancia resaltar que la etapa de enriquecimiento del conocimiento se integra en ambos ciclos, y la etapa de discusión del modelo los conecta intrincadamente. Además, en ambos ciclos se llevan a cabo de manera sistemática actividades de reflexión y validación.

1. Evaluación del DCEI y de las necesidades de intervención.
2. Toma de decisiones facilitada por herramientas del PS.
3. Encuadre y conceptualización de la problemática, seguido de la discusión y validación de modelos.
4. Reflexión y verificación recurrentes para garantizar la precisión y relevancia de la información y los modelos.
5. Enriquecimiento del conocimiento para respaldar la toma de decisiones.
6. Análisis exhaustivo de datos e información para reducir incertidumbres y optimizar recursos.
7. Generación de modelos para representar la estructura y dinámica del dominio.
8. Evaluación de la idoneidad de las soluciones propuestas.

4. La GC y PS en el marco KMoS-SSA

El marco KMoS-SSA promueve un abordaje holístico del DCEI, considerando tanto elementos estructurados como no estructurados y dimensiones sociales y culturales, destacando:

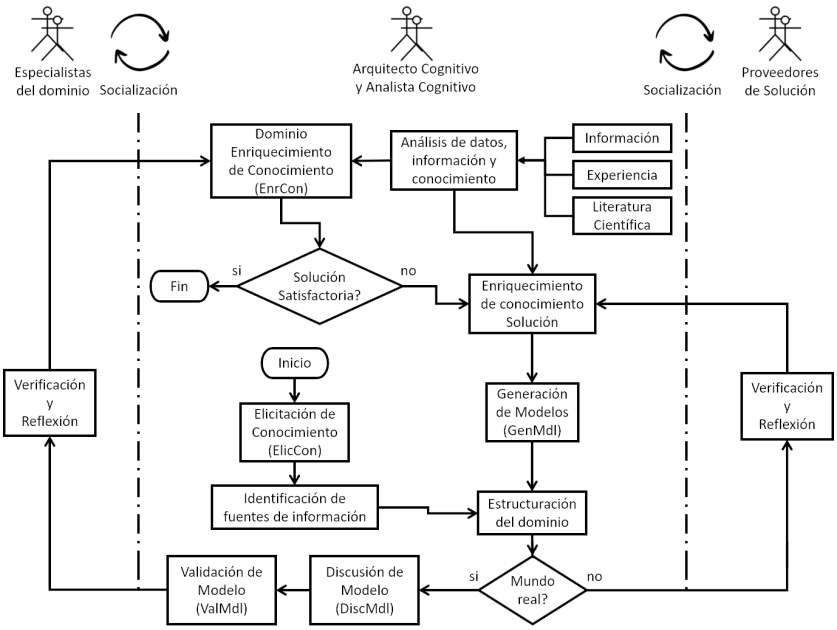


Fig. 3. corresponde a una representación del marco KMoS-SSA, visualizando las etapas y actividades fundamentales para la GC. Los cuadros rectangulares denotan las etapas y sus respectivas actividades, mientras que los rombos simbolizan las decisiones. Los puntos “inicialc “final”se señalan con elipsis. En la parte superior, las figuras humanas representan los principales actores de un DCEI, así como sus interacciones.

1. Análisis de Problemas y Necesidades: Análisis exhaustivo de los problemas y necesidades presentes dentro del ámbito definido, utilizando herramientas como entrevistas, encuestas y análisis de documentos para recopilar información y conocimientos sobre los desafíos existentes.
2. Reconocimiento de las Interconexiones: Se analizan las interconexiones entre los diferentes actores, elementos o componentes desde la perspectiva sistémica, identificando cómo los cambios en alguno de estos elementos pueden tener repercusiones en otros.
3. Obtención de Requisitos: Se emplean entrevistas, talleres y cuestionarios para recopilar requisitos de los responsables de decisiones y actores clave del DCEI, con énfasis en comprender sus expectativas y necesidades.
4. Análisis de Contingencias y Escenarios: Se exploran diversos escenarios y situaciones para comprender las contingencias y posibles variaciones en el comportamiento del dominio, identificando requisitos que aborden todos los escenarios potenciales dentro del DCEI.
5. Modelización del DCEI a través del Conocimiento del Dominio: Se recurre a herramientas de modelado para representar visualmente las piezas de conocimiento, requisitos y otros elementos pertinentes mediante imágenes, descripciones

documentadas, diagramas de casos de uso, diagramas de flujo y modelos conceptuales. El propósito radica en mejorar la comunicación y asegurar la comprensión entre todos los actores del dominio.

6. Documentación de la Adquisición de Elementos de Conocimiento y su Impacto en el DCEI: Se elabora una documentación exhaustiva que abarca todas las prioridades y restricciones derivadas de las preferencias de los actores dentro del DCEI, especialmente de los tomadores de decisiones.

5. Discusión y trabajo futuro

El presente artículo destaca que el marco KMoS-SSA proporciona soluciones prácticas a instituciones, organizaciones o empresas que buscan mitigar los impactos de los problemas dentro de un DCEI. Específicamente, en el ámbito del estrés laboral, donde están emergiendo los primeros resultados positivos del KMoS-SSA [12], el marco contribuye significativamente al diálogo en curso sobre estrategias eficaces de gestión de este DCEI en particular. En los últimos 10 años, el KMoS-SSA ha resuelto con éxito dos problemas por año, alcanzando una tasa de satisfacción del 95.243 %. Ha abordado una amplia gama de áreas complejas, desde la atención médica hasta la logística, mediante el desarrollo de herramientas y sistemas para mejorar la cognición humana y la eficiencia de procesos.

El enfoque del KMoS-SSA en requisitos y representación del conocimiento ha permitido desarrollar soluciones innovadoras para problemas del mundo real, como la prevención de enfermedades crónicas y la optimización logística. Es importante recordar que el antecedente de KMoS-SSA, KMoS-RE, se ha centrado en el uso de técnicas de análisis de requisitos, enriquecidas con técnicas de psicología y comunicación, para el diseño de soluciones cognitivas, especialmente en un DCEI. En este sentido, el KMoS-SSA ha permitido definir requisitos de conocimiento y obtener las piezas de conocimiento necesarias para el desarrollo de las soluciones.

Ha demostrado ser particularmente útil en DCEI, donde la definición de componentes y la determinación de piezas de conocimiento se abordan de manera colaborativa entre todos los actores del dominio. Las soluciones convencionales pueden ser insuficientes para abordar las intrincadas problemáticas presentes en los DCEI, lo que subraya la importancia de adoptar enfoques innovadores y adaptables, incluida la integración de tecnologías cognitivas como algunas herramientas de IA. En el caso del estrés laboral, un desafío clave es gestionar eficazmente el vasto conocimiento, tanto tácito como implícito, necesario para abordar sus diversos aspectos.

Históricamente, tanto las tácticas sistémicas como las de gestión del conocimiento han enfrentado dificultades cuando se aplican por separado en dominios complejos, como la gestión ambiental y empresarial [10, 21], las problemáticas sociales y de salud mental [32, 28], y la sostenibilidad [31], entre otros. En consecuencia, estos desafíos persisten debido a la complejidad inherente de tales asuntos. La convergencia y armonización de perspectivas, cosmovisiones y prácticas de gestión del conocimiento, entre otros factores, presentan complejidades que demandan la formulación de filosofías o enfoques integrados para abordar eficazmente la complejidad de los DCEI.

En este sentido, el marco KMoS-SSA emerge como una opción idónea para la gestión del conocimiento con un enfoque sistémico. En su evolución actual, en 2024, el KMoS-SSA se ha fortalecido con la inclusión de enfoques sistémicos, específicamente las intervenciones de los métodos de sistemas flexibles, lo que ha generado un nicho de oportunidad interesante para abordar situaciones del mundo real que se caracterizan por complejas interacciones, diversas perspectivas y la gestión de relaciones de poder en el contexto de una EC. Esta inclusión tiene como objetivo principal facilitar y adoptar la gestión sistemática del conocimiento. Una situación interesante que ejemplifica estos conceptos es el DCEI del estrés laboral, el cual presenta una problemática intrincada que se ve exacerbada en la EC y reviste una importancia sustancial debido a sus consecuencias adversas para la salud pública, la economía, las organizaciones y la sociedad en general.

El estrés laboral, un dominio complejo caracterizado por la diversidad de la fuerza laboral, la participación de especialistas de diversas disciplinas y la dinámica organizacional, halla en el marco KMoS-SSA una solución práctica para mitigar sus repercusiones dentro del contexto de la innovación empresarial. Con un enfoque sistémico robustecido, este marco está demostrando generar resultados positivos en la gestión del estrés laboral, aportando de manera significativa al diálogo en torno a estrategias efectivas para su abordaje.

Entre estas estrategias se incluye el desarrollo de un juego serio fundamentado en la teoría de Karasek [13] para la sensibilización sobre el estrés laboral [16], la implementación de una ontología del estrés laboral [24], y actualmente se encuentra en curso la elaboración de un sistema recomendador basado en grafos de conocimiento para la gestión del estrés laboral en el sector de la industria maquiladora [2]. Los dominios complejos, como el estrés laboral, son una parte intrínseca de la vida cotidiana, extendiéndose más allá de casos específicos a contextos donde las personas utilizan conocimientos para tomar decisiones, resolver problemas y satisfacer necesidades. Así, herramientas como el KMoS-SSA son esenciales para abordar eficazmente estos dominios, ya sea a nivel industrial, institucional o en cualquier organización que experimente procesos de transformación cognitiva.

A pesar de los logros obtenidos, hay espacio para mejoras, por lo que se están desarrollando tres tesis doctorales para ampliar aún más el alcance y la eficacia del marco KMoS-SSA en la resolución de problemas complejos. Una de estas tesis se centra en el desarrollo de un metaproceso para la implementación de una arquitectura cognitva ad hoc [14]. Este metaproceso supervisaría y coordinaría todas las etapas del proceso, incluyendo la planificación estratégica, el análisis y diseño, el desarrollo, la implementación, la evaluación, la optimización y el mantenimiento, con el fin de garantizar que la arquitectura cognitiva avance de manera eficiente y se alcancen los objetivos establecidos [12]. Otra de las tesis se enfoca en investigar técnicas de representación de piezas de conocimiento derivadas del conocimiento tácito, como es el caso del conocimiento geométrico [5]. Finalmente, uno de los desafíos más destacados que plantea un DCEI es el relacionado con la transferencia de conocimiento.

Aunque partimos de una fuerte influencia del modelo SECI [6], aún queda mucho por hacer en cuanto a la formalización de un proceso de negociación y confianza basado en el afecto y en la cognición de los especialistas del dominio y otros actores relevantes

del DCEI, con el fin de lograr compartir y utilizar eficazmente el conocimiento tácito [11]. En conclusión, este artículo ha expuesto el marco KMoS-SSA como una herramienta efectiva para abordar dominios complejos, al integrar enfoques adaptables para gestionar la complejidad inherente a la transformación digital y cognitiva de las organizaciones. Al adoptar este marco, es posible mejorar la toma de decisiones, fomentar la innovación y enfrentar de manera más efectiva los desafíos contemporáneos que surgen en diversos dominios complejos.

Referencias

1. Aviad, A. E., Wecel, K., Abramowicz, W.: A semantic approach to modelling of cybersecurity domain. Journal of Information Warfare, vol. 15, no. 1, pp. 91–102 (2016)
2. Bretado-Retana, J. H: Support system for organizational decision making of work stress using knowledge graphs. Memorias Científicas y Tecnológicas, vol. 2, no. 1, pp. 18 (2023)
3. Checkland, P. B.: Soft systems methodology. Human Systems Management, vol. 8, no. 4, pp. 273–289 (1989) doi: 10.3233/hsm-1989-8405
4. Checkland, P., Poulter, J.: Soft systems methodology. Systems Approaches to Making Change: A Practical Guide, pp. 201–253 (2020) doi: 10.1007/978-1-4471-7472-1_5
5. de-Mello, F. L., de-Carvalho, R. L.: Knowledge geometry. Journal of Information and Knowledge Management, vol. 14, no. 4, pp. 1550028 (2015) doi: 10.1142/s02196492 15500288
6. Farnese, M. L., Barbieri, B., Chirumbolo, A., Patriotta, G.: Managing knowledge in organizations: a Nonaka's SECI model operationalization. Frontiers in Psychology, vol. 10, pp. 2730 (2019) doi: 10.3389/fpsyg.2019.02730
7. Forrester, J. W.: Industrial dynamics. MIT Press (1961)
8. Hadi, A. H., Suprihatin, Sukardi, Pramuhadi, G., Marimin, Susantyo, B., Wahyono, E.: Sustainability concept design of robusta coffee agroindustry Kalibaru with soft system and decisions support system methods. International Journal of Sustainable Development and Planning, vol. 18, no. 5, pp. 1339–1350 (2023) doi: 10.18280/ijsdp.180504
9. Hanafizadeh, P., Mehrabioun, M.: The nature of hard and soft problems and their problemsolving perspectives. Journal of Systems Thinking in Practice, vol. 1, no. 3, pp. 22–48 (2022) doi: 10.22067/JSTINP.2022.79419.1024
10. Hassan, S.: Soft systems methodology in environment-aware case-based reasoning system analysis. Information Technology Journal, vol. 9, no. 3, pp. 467–473 (2010) doi: 10.3923/itj.2010.467.473
11. Holste, J. S., Fields, D.: Trust and tacit knowledge sharing and use. Journal of Knowledge Management, vol. 14, no. 1, pp. 128–140 (2010) doi: 10.1108/13673271011015615
12. Jiménez-Galina, A. M., Maldonado-Macias, A. A., Olmos-Sánchez, K. M., Licona-Olmos, J. G.: Necesidad de una metodología de representación del conocimiento para sistemas complejos en dominios de estructura informal: Artículo sobre enfoques empírico-teóricos. In: Proceedings of the 11th International Conference on Software Engineering Research and Innovation, Sesión B2 (2023)
13. Karasek, R. A.: Job demands, job decision latitude, and mental strain: Implication for job redesign. Administrative Science Quarterly, vol. 24, no. 2, pp. 285–308 (1979)
14. Kotseruba, I., Tsotsos, J. K.: 40 years of cognitive architectures: core cognitive abilities and practical applications. Artificial Intelligence Review, vol. 53, no. 1, pp. 17–94 (2018) doi: 10.1007/s10462-018-9646-y
15. Larson, E. J.: The myth of artificial intelligence: Why computers can't think the way we do. Harvard University Press (2021) doi: 10.2307/j.ctv322v43j

16. Maldonado-Macías, A. A., Olmos-Sanchez, K. M., Jiménez-Galina, A. M., Pereyra-Manriquez, J. D.: Serious game prototype for burnout awareness among maquiladora employees in Ciudad Juárez. Ergonomía Ocupacional, Investigaciones y Aplicaciones, vol. 15, pp. 341–355 (2022)
17. Nakamori, Y.: Fusing systems thinking with knowledge management. Journal of Systems Science and Systems Engineering, vol. 29, no. 3, pp. 291–305 (2020) doi: 10.1007/s11518-019-5450-8
18. Nikhlis, N., Iriani, A., Hartomo, K. D.: Soft system methodology (SSM) analysis to increase the number of prospective students. INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi, vol. 4, no. 1, pp. 63–74 (2020) doi: 10.29407/intensif.v4i1.13552
19. Nonaka, I., Takeuchi, H.: The knowledge-creating company: How japanese companies create the dynamics of innovation. Hardvard Business Review, pp. 96–104 (1995)
20. Olmos-Sanchez K., Rodas-Osollo, J.: A strategy of requirements engineering for informally structured domains. International Journal of Combinatorial Optimization Problems and Informatics, vol. 7, no. 2, pp. 49–56 (2016)
21. Olmos-Sanchez, K., Rodas-Osollo, J.: Helping organizations manage the innovation process to join the cognitive era. In: Proceedings of the 8th International Conference in Software Engineering Research and Innovation, pp. 1–10 (2020) doi: 10.1109/conisoft50191.2020.00012
22. Organización Internacional del Trabajo: Gestión de los riesgos psicosociales relacionados con el trabajo durante la pandemia de COVID-19. pp. 36 (2020)
23. Polanyi, M.: Personal knowledge. University of Chicago, 2nd Edition, pp. 492 (1958)
24. Pérez-Campos, A. P., Olmos-Sánchez, K. M., Maldonado-Macias, A., Jiménez-Galina, A. M.: Desarrollo de una ontología de estrés laboral orientada la industria maquiladora en la frontera de México. In Memorias Científicas y Tecnológicas, vol. 2, no. 1, pp. 56–57 (2022)
25. Reynolds, M., Holwell, S.: Systems approaches to making change: A practical guide. Springer London, pp. 201–253 (2020) doi: 10.1007/978-1-4471-7472-1
26. Rodas-Osollo, J., Olmos-Sánchez, K., Kotlyarova, I., Jimenez-Galina, A.: KMoS-SSA: Gestión sistémica del conocimiento. In: Congreso Mexicano de Inteligencia Artificial (2024)
27. Rodas-Osollo, J., Olmos-Sánchez, K., Portillo-Pizaña, E., Martínez-Pérez, A., Alemán-Meza, B.: An archetype of cognitive innovation as support for the development of cognitive solutions in smart cities. Innovative Applications in Smart Cities, pp. 89–105 (2021) doi: 10.1201/9781003191148-8
28. Rodas-Osollo, J., Olmos-Sánchez, K.: Toward optimization of medical therapies with a little help from knowledge management. Recent Advances in Knowledge Management (2022) doi: 10.5772/intechopen.101987
29. Rodas-Osollo, J.: An interesting adventure accompanied by CMCg.I model. Zenodo (2023) doi: 10.5281/zenodo.10111223
30. Rohajawati, S., Fairus, S., Saragih, H., Akbar, H., Rahayu, P.: A combining method for systems requirement of knowledge - based medical hazardous waste. TEM Journal, vol. 10, no. 4, pp. 1761–1768 (2021) doi: 10.18421/tem104-37
31. Tona, O., Asatiani, A.: Designing digital solutions for sustainability: Navigating conflicting stakeholder requirements with dignity in mind. Journal of Information Technology Teaching Cases, pp. 1–8 (2023) doi: 10.1177/20438869231216995
32. Trochim, W. M., Cabrera, D. A., Milstein, B., Gallagher, R. S., Leischow, S. J.: Practical challenges of systems thinking and modeling in public health. American Journal of Public Health, vol. 96, no. 3, pp. 538–546 (2006) doi: 10.2105/ajph.2005.066001
33. Zahid, A., Sharma, R., Wingreen, S., Inthiran, A.: Soft systems modelling of design artefacts for blockchain-enabled precision healthcare as a service. In: Proceedings of the International Conference on Electronic Business, vol. 22, pp. 451–467 (2022)

ISSN 1870-4069

Estudio del rol que desempeña el big data en las políticas públicas que impactan en la urbanización sustentable agua-vivienda

José Luis Hernández-González, Rodolfo Eleazar Pérez-Loaiza, Perfecto Malaquías Quintero-Flores

Tecnológico Nacional de México,
División de Estudios de Posgrado e Investigación,
México

{luis.hg, rodolfo.pl, perfecto.qf}@apizaco.tecnm.mx

Resumen. Este análisis exploratorio investiga los problemas emergentes de los gobiernos locales, regionales, nacionales e internacionales asociados al big data. Se describen los ejes rectores propuestos por la ONU, su inclusión en el plan nacional de desarrollo y su abordaje desde el big data. Se exploran de forma no exhaustiva dos problemáticas del big data urbano: Agua y Vivienda, destacando la implicación del big data en el problema del agua y la posible falta de datos abiertos.

Palabras clave: Agua, sustentable, big data.

Study of the Role that Big Data Plays in Public Policies that Impact Sustainable Water-housing Urbanization

Abstract. This exploratory analysis investigates emerging issues faced by local, regional, national, and international governments related to big data. It outlines the guiding principles proposed by the UN, their inclusion in the national development plan, and their approach through big data. Two urban big data issues, Water and Housing, are explored in a non-exhaustive manner, highlighting big data's implications for water issues and the potential lack of open data.

Keywords: Water, sustainable, big data.

1. Introducción

En el presente artículo, se discute el rol del big data en la toma de decisiones y las problemáticas derivadas de los asentamientos urbanos y su implicación en las políticas públicas; asimismo, se delimitarán dos conceptos considerados emergentes: el agua y la vivienda. El Plan Nacional de Desarrollo del gobierno federal, ha definido cuatro ejes rectores, los cuales se encuentran inmersos en problemáticas mundiales orientadas principalmente al agua, al desarrollo sostenible, a la suficiencia alimenticia, a la vivienda, a la movilidad, entre otras.

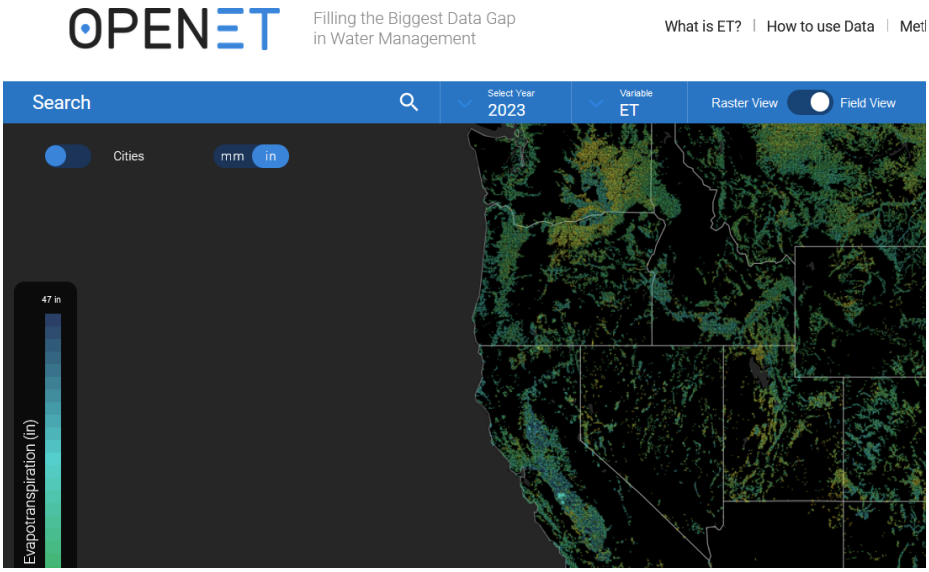


Fig. 1. Plataforma OpenET datos de evotraspiración. Imagen tomada de [6].

En la actualidad, el crecimiento en la cantidad de datos asociados a esas problemáticas ha generado un alto interés a la inclusión del denominado big data urbano y su posible incorporación en tales políticas públicas. Se ha vislumbrado un campo de trabajo asociado a la búsqueda de estrategias inteligentes que van desde recuperar, organizar, procesar, analizar e interpretar grandes volúmenes de datos, por lo que se requiere establecer metodologías que permitan dar solución de manera eficiente a las problemáticas urbanas; algunos países incluyendo México, se encuentran en la necesidad de fortalecer una política de datos abiertos que permitan a la ciudadanía conocer el estado actual de proyectos emergentes, la posibilidad de extraer y analizar información para realizar la toma de decisiones como por ejemplo, la atención de falta de agua para el consumo, su uso en la agricultura, formas de tratamiento de las aguas residuales entre otros.

2. Políticas públicas

Parte fundamental del quehacer político del país, se encuentra enmarcado en los diferentes planes rectores que van desde el ámbito nacional, regional o local para atender los problemas nacionales de acuerdo con los planes de trabajo de cada gobierno; para ello, es necesario conocer las políticas públicas asociadas a las problemáticas emergentes para atención a la sociedad. Se presenta brevemente la visión o interés en atender problemas asociados a entes internacionales e identificar cuáles son las que ha establecido atender el gobierno actual. La Organización de Naciones Unidas (ONU) es una organización internacional conformada por 193 estados y fue fundada en 1945 con la finalidad de mantener la paz y la seguridad internacional para solucionar problemas globales.

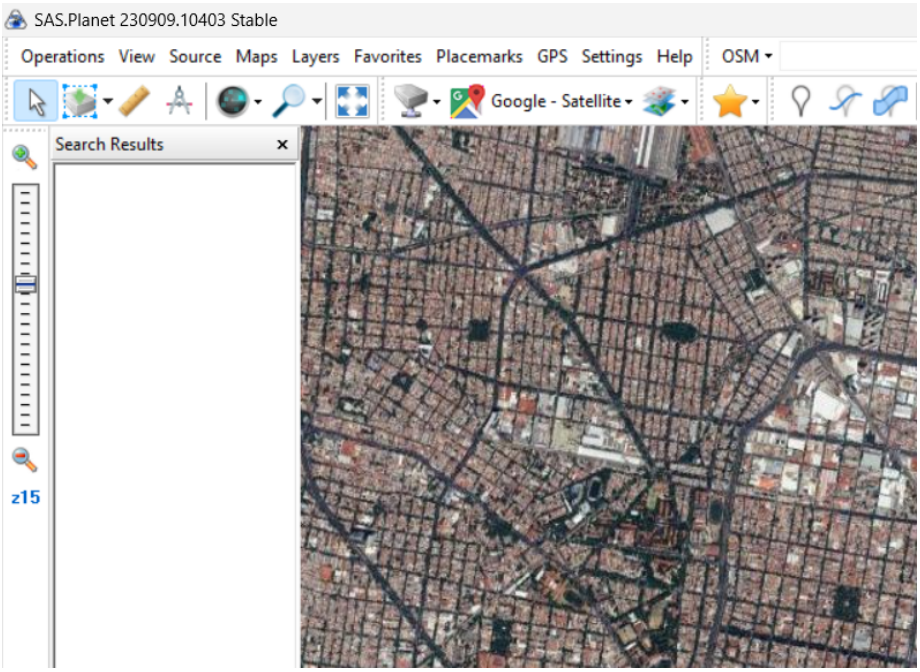


Fig. 2. Imagen satelital, tomada de la Plataforma SASGIS [10].

Actualmente, también incluye la promoción y protección de los derechos humanos, el desarrollo sostenible y la cooperación internacional en economía, problemática social, cultura y acciones humanitarias. El eje rector es la "Carta de las Naciones Unidas", que es considerado un tratado internacional y recoge los principios de las relaciones internacionales desde la igualdad hasta la prohibición de la fuerza en las relaciones internacionales. La carta se encuentra constituida por XIX capítulos y 111 artículos. México es miembro desde su admisión el 7 de noviembre de 1945 [7]. Las actividades de la ONU se encuentran definidas en cinco temáticas principales:

1. Mantener la paz y la seguridad internacionales.
2. Proteger los derechos humanos.
3. Distribuir ayuda humanitaria.
4. Apoyar el Desarrollo sostenible y la acción climática.
5. Defender el derecho internacional.

Además, se encuentran inmersos en 22 desafíos globales, los cuales deben atenderse de manera conjunta ya que un solo país no puede resolverlos de forma individual. Uno de los desafíos globales asociados al big data son: Macrodatos para el Desarrollo sostenible. La ONU menciona que los macrodatos, es un área de oportunidad que se encuentra emergiendo con el vertiginoso crecimiento de la informática y las nuevas tecnologías, datos y su posible análisis.

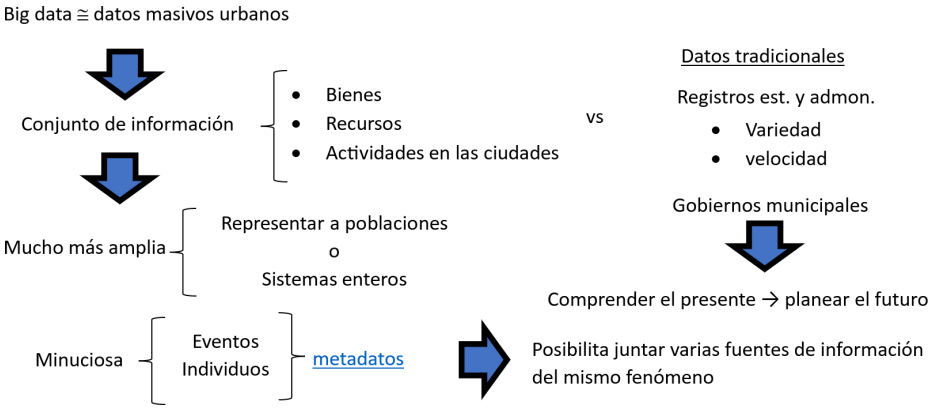


Fig. 3. Fuentes de información urbana.

Con ello, se mejorará el control de los Objetivos del Desarrollo Sostenible (ODS), los desafíos globales se encuentran definidos en la agenda 2030. La agenda 2030 es la aprobación de 17 ODS, firmada por todos los estados miembros en el año 2015, tal agenda establece un plan para la acción y poner fin a la pobreza, cuidado del planeta, mejorar la vida y ha sido denominada la Agenda 2030 para el Desarrollo Sostenible:

1. Fin de la pobreza.
2. Hambre cero.
3. Salud y bienestar.
4. Educación de calidad.
5. Igualdad de género.
6. Agua limpia y saneamiento.
7. Energía accesible no contaminante.
8. Trabajo decente y crecimiento económico.
9. Industria, innovación e infraestructura.
10. Reducción de la desigualdad.
11. Ciudades y comunidades sostenibles.
12. Producción y consumo responsable.
13. Acción por el clima.
14. Vida submarina.
15. Vida de ecosistemas terrestres.
16. Paz y justicia e instituciones sólidas.
17. Alianza para lograr los objetivos.

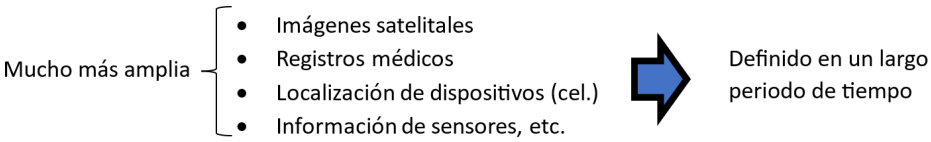


Fig. 4. Acciones humanas recogidas por dispositivos digitales.

El objetivo 6: Agua limpia y saneamiento es una problemática de suma importancia para la sociedad, de acuerdo con la ONU en el 2030 miles de millones de personas no tendrán acceso a agua potable que se requiere para su consumo, el cuidado de la salud y el bienestar. El objetivo 11: Lograr que las ciudades sean más inclusivas, seguras, resilientes y sostenibles, así como el objetivo 17: Revitalizar la alianza mundial para el desarrollo sostenible, son el fundamento de análisis para establecer una estrategia que permita abordar el big data respecto al agua y la vivienda. El ODS 17 menciona 2 metas a atender: Meta 17.18 aumentar significativamente la disponibilidad de datos oportunos y la Meta 17.19 aprovechar las iniciativas existentes para elaborar indicadores y, son relevantes para la búsqueda de soluciones en dos temáticas de interés en el futuro para las ciudades inteligentes: agua y vivienda.

México ha adoptado la agenda 2030 y los ODS, el artículo 25 de la constitución política menciona que: "Corresponde al Estado la rectoría del desarrollo nacional para garantizar que éste sea integral y sustentable, que fortalezca la Soberanía de la Nación y su régimen democrático y que, mediante el fomento del crecimiento económico y el empleo y una más justa distribución del ingreso y la riqueza, permita el pleno ejercicio de la libertad y la dignidad de los individuos, grupos y clases sociales, cuya seguridad protege esta Constitución . . . " [2]. Para ello, el Plan Nacional de Desarrollo 2019-2024 ha definido 4 ejes rectores [11]:

1. Política y Gobierno.
2. Política Social.
3. Economía.
4. Epílogo: Visión del 2024.

De los cuales, se desprenden 9 programas:

1. El programa para el bienestar de las personas adultas mayores.
2. El Programa Pensión para el Bienestar de las Personas con Discapacidad.
3. El Programa Nacional de Becas para el Bienestar Benito Juárez.
4. Jóvenes Construyendo el Futuro.
5. Jóvenes escribiendo el futuro.
6. Sembrando vida.
7. Programa Nacional de Reconstrucción.
8. Desarrollo Urbano y Vivienda.
9. Tandas para el bienestar.

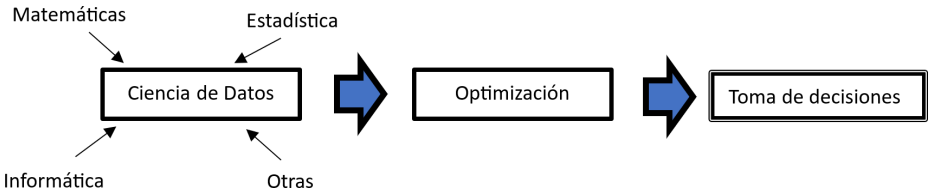


Fig. 5. Una primera aproximación de ciencia de datos.

3. Problemática

Determinar algunas de las problemáticas asociadas al Plan Nacional de Desarrollo vigente e identificar el nivel desarrollo asociado al eje rector política social, en particular al programa ocho, desarrollo urbano y vivienda. Recuperando el ODS 11 ciudades y comunidades sostenibles, el informe de la agenda 2030 en México "Data para el Desarrollo sostenible", se presentan estadísticas referentes al crecimiento urbano. Algunas conclusiones son:

- Más de la mitad de la población mundial reside actualmente en zonas urbanas.
- El crecimiento urbano no es controlado, la contaminación atmosférica y la falta de espacios públicos abiertos no han sido subsanados.
- Se requieren políticas y prácticas de desarrollo urbano inclusivo, resiliente y sostenible, hacen falta áreas verdes y mejorar transporte público.
- La expansión urbana supera el crecimiento poblacional a escala global.
- El problema de agua y vivienda debe ser atendido de manera prioritaria.

Existen una gran cantidad de problemas asociados con el uso del agua, que van desde la cantidad destinada al consumo humano, la protección de las fuentes de agua, la captación del agua, la contaminación del agua, uso del agua para la agricultura, entre muchos otros. Se requiere evaluar el consumo en la cantidad de agua disponible y la relación que existe con el crecimiento urbano.

Diferentes autores como Breña [1] indican que ya es posible la inclusión tecnológica en el sector hídrico, con la finalidad de que cualquier persona pueda beneficiarse con la información digital y aunque es accesible para todos, no se cuenta con herramientas disponibles que permitan desmenuzar la información, existen plataformas que nos muestran a través de mapas cartográficos con la diversa información, pero solamente personas con conocimientos especializados la pueden interpretar.

Por otro lado, existen diferentes tipos de formatos para la información tanto libre como privada, a continuación, se presentan algunas estrategias del uso del big data asociado al agua y Desarrollo Urbano o Vivienda. Como lo indica Breña, ya se cuenta con información georreferenciada útil para atender problemáticas como la agricultura. La Figura 1 muestra la Plataforma web OpenET a la que se refiere Breña, la cual proporciona datos diarios, mensuales o anuales de evotraspiración para el uso del agua al este de Estados Unidos [6].

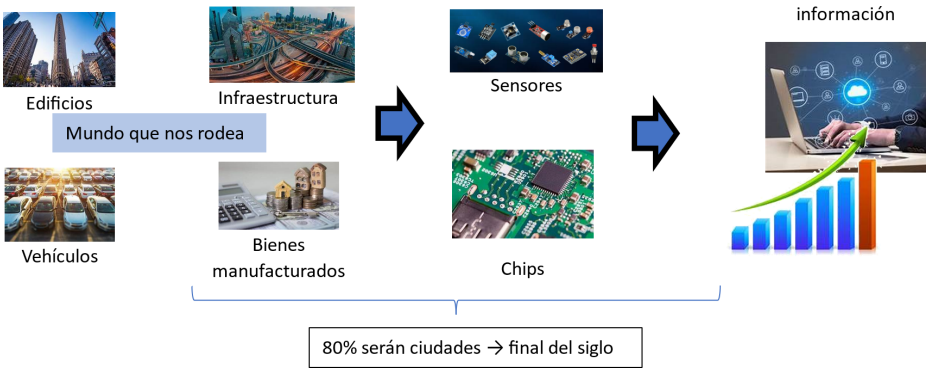


Fig. 6. Representación del big data. Imágenes tomadas de internet.

Del mismo modo, la figura 2 muestra una parte de una imagen satelital que permitirá mediante alguna metodología, proporcionar información apropiada para medir por ejemplo crecimiento urbano, identificar vías de comunicación, establecer rutas, entre algunas otras variables de interés para la toma de decisiones en el sector urbano [10].

Debido a que una de las problemáticas de las ciudades radica principalmente en el tiempo que se consume en realizar actividades cotidianas tales como realizar el traslado del hogar al trabajo, a las escuelas, a los mercados, a los hospitales etc., se requiere identificar o mejorar las rutas para optimizar tiempo y dinero tanto para la población en general como trabajadores del sector gubernamental y privado. El Banco Interamericano del Desarrollo (BID) ha financiado proyectos orientados al big data y el desarrollo urbano, uno de los resultados es "Urban Reporting based on Satellite Analysis" (URSA) el cual, es un sistema abierto que permite acceder de forma sencilla a la información digital de sensores satelitales URSA [13].

También se cuenta con información en formato numérico como datos estadísticos en determinado período de tiempo, por lo que se requiere de otras estrategias de la ciencia de datos para analizar, interpretar, presentar y proyectar información al futuro para la toma de decisiones, el INEGI, cuenta con bases de datos con identificadores con las características de la población, hogares censados y las viviendas, algunas de las variables de interés son: población por su edad, características económicas, servicios de salud, servicios de que se dispone (luz, agua, drenaje), para esta propuesta se tomará la información del censo de población y vivienda 2020 y hacer un análisis descriptivo [5].

4. Conceptualización de big data urbana

La figura 3 muestra cómo es que, a través de la convivencia urbana y el uso de dispositivos actuales, se genera una gran cantidad de datos, la ciudadanía genera información de sus actividades cotidianas, hace consultas, por ejemplo, precios, acceso a la banca digital, recibe información indirecta en sus celulares, que es de mayor variabilidad que la obtenida por registros tradicionales y mucho más amplia e incluye metadatos [12].

Año	Población
1950	25,791,017
1960	34,923,129
1970	48,225,238
1980	66,846,833
1990	81,249,645
1995	91,158,290
2000	97,483,412
2005	103,263,388
2010	112,336,538
2020	126,014,024

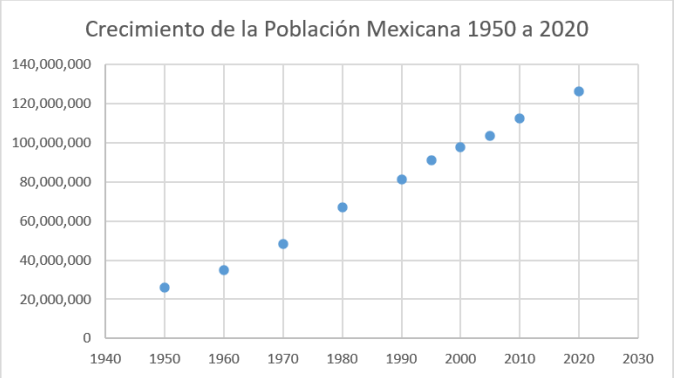


Fig. 7. Crecimiento de la población mexicana de 1950 a 2020.

La figura 4, muestra algunas de las acciones que permiten recolectar grandes cantidades de información y son consideradas como big data urbana. Con el uso de los satélites artificiales se produce información en sus diferentes tipos desde información geográfica en formato Gis o Rastrer o imágenes en formatos gif o jpeg; en el caso de la medicina aunque algunos datos son producidos de forma manual, otras son a través de dispositivos como baumanómetros y termómetros digitales, toma de muestras de sangre con glucómetros y diferentes estudios especializados como ultrasonidos en sus diferentes tipos, radiografías, etc. producen datos, pero de manera sigilosa los celulares, relojes digitales y otros dispositivos registran acciones del usuario, además de la información que generan los sensores incorporados y que pueden ser lecturas en determinados intervalos o por tiempo indefinido.

Dentro de las posibles definiciones de big data, se encuentran aquellas que mencionan que los grandes conjuntos de datos cumplen con las siguientes tres características: gran cantidad de datos, variabilidad en su forma y velocidad para su generación, de aquí que se presenta una complejidad en su análisis; por lo cual, se requiere explorar tales conjuntos de datos, su integración con variables de interés definidas y la exploración de diversas estrategias para su tratamiento.

La ciencia de datos, de acuerdo con la figura 5 muestra una primera aproximación de su uso, la cual se apoya de otras disciplinas como las matemáticas, la estadística, la informática, entre otras, para optimizar y dar soluciones que permitan realizar la toma eficiente de decisiones [12]. Actualmente, el sector privado, el sector público y el gobierno se encuentran haciendo alianzas para generar estrategias para tratar las grandes cantidades de datos que se producen en las ciudades, lo que ha propiciado que algunos expertos ya han acuñado el término de big data urbana.

La figura 6 muestra la visión del big data urbano, de acuerdo con lo expuesto por Townsend y Zambrano [12]. Ellos mencionan que el mundo que nos rodea, nuevos edificios inteligentes, sensores en la infraestructura para la medición de velocidad, toma de fotografías, audio y video, medición de humedad, de contaminantes, de temperatura ambiental, uso de los medios de transporte como la geolocalización, y otras acciones de las actividades de carácter económico, financieras, comerciales incrementan la cantidad de datos.

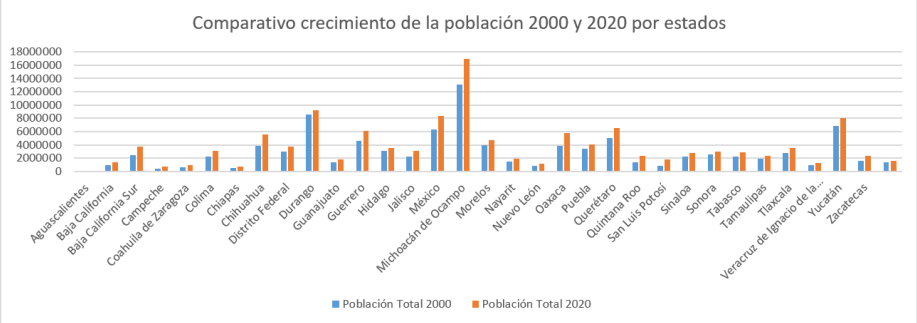


Fig. 8. Comparativo crecimiento de la población 2000 y 2020 por estados.

5. Resultados del análisis exploratorio

Para la propuesta de este trabajo, se ha realizado la búsqueda de datos abiertos con la finalidad de identificar problemáticas a atender con los Pronaces (Programas Nacionales Estratégicos) Agua y Vivienda; de manera preliminar, se ha seleccionado la base de datos de resultados obtenidos del Censo de Población y Vivienda 2020, para realizar un análisis descriptivo, de manera adicional, se hará uso de los censos anteriores para evaluar el crecimiento de consumo de agua tanto de los censos 2005, 2010, 2015, además, del 2020. De acuerdo con Padilla [8] y según recomendación de la Organización Mundial de Salud una persona necesita 100 litros de agua al día, por lo que se podrá estimar la cantidad de agua consumida a nivel nacional, por estado o municipio según la encuesta de INEGI.

Las gráficas que se muestran, la población total con la finalidad de establecer la cantidad de agua requerida por habitante. La gráfica en la figura 7 muestra el crecimiento de la población de los años 1950 al 2020 de acuerdo a la información de los censos y conteos de población y vivienda, se incluyen 2 conteos. El primer censo se realizó en 1895. La gráfica en la figura 8 muestra un comparativo de la población por estados para los años 2000 y 2020, se han realizado también las comparaciones con los años 2005 y 2010 con la finalidad de identificar qué estados requieren más cantidades de dotación de agua. La gráfica en la figura 9 muestra la distribución espacial de la población total por estado, para el análisis exploratorio descriptivo se elaboraron gráficas de los cuartiles, se exploran deciles y percentiles con la finalidad de observar a detalle los cambios del crecimiento por estado.

Para un primer análisis exploratorio, se ha seleccionado la variable población total con la finalidad de determinar la cantidad de consumo requerido por habitante/día, sin embargo, Ramos [9] en su presentación “Análisis socio-espacial del uso doméstico del agua en la ciudad de México: Hacia la gestión integrada del agua urbana”, presenta datos estadísticos de la situación del agua en la ciudad de México, menciona tres tipos de uso del agua: doméstico, no doméstico y mixto; para ello, define las siguientes variables independientes: porcentaje de usuarios domésticos, población total de la colonia, tamaño promedio del hogar, Índice de Desarrollo Social (IDS) promedio, densidad de viviendas por colonia y abasto por tandeo, para el caso de la

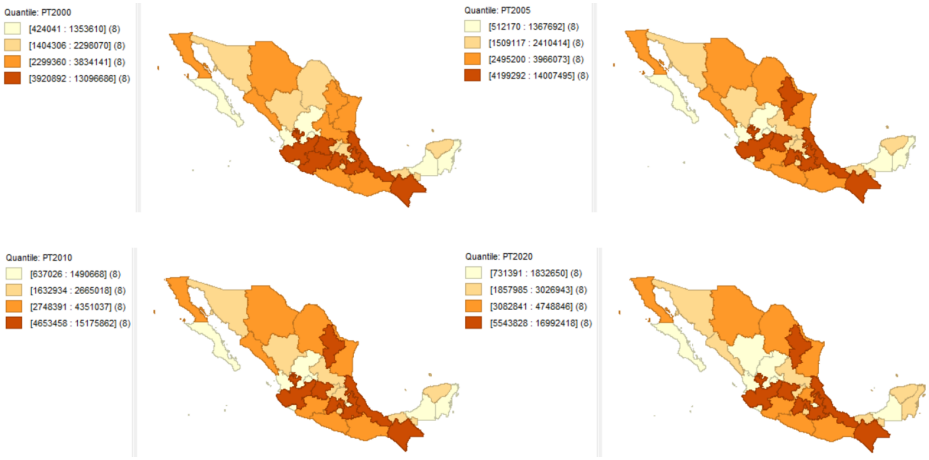


Fig. 9. Distribución espacial de la población 2000, 2005, 2010, 2020.

variable dependiente: uso doméstico en forma bruta (m3/año), uso doméstico per cápita (l/hab/día), mismos que dependiendo de la fuente de los datos serán incorporadas en esta propuesta.

6. Conclusiones

Algunos países han regulado sus políticas para la incorporación de datos abiertos con la finalidad de modernizar la administración pública y consolidar la transparencia y eficiencia; en el caso del gobierno de México, ha publicado la Política de Transparencia, Gobierno Abierto y Datos Abiertos de la administración pública federal 2021-2024, cuya finalidad es contribuir a la rendición de cuentas y combate a la corrupción e impunidad. Una de las áreas de oportunidad, es la de datos abiertos: mejorar la disponibilidad de datos publicados por el gobierno con las siguientes características: existencia de datos, formato digital estructurado, acceso gratuito, licencia abierta.

En la política de transparencia, se definen tres ejes de acción, en cada eje se definen seis prioridades con la finalidad de desarrollar acciones para consolidar la transparencia, el gobierno abierto y los datos abiertos. En el eje 3: Impulsar el uso de datos abiertos, con prioridad 6, se menciona que se deberán: “Implementar procesos de apertura de datos abiertos que sean de mayor interés o relevancia para la población, con la finalidad de contribuir al bienestar social” [4].

Se requiere establecer metodologías que permitan extraer la información de pertinencia y proporcionar información para realizar la toma de decisiones; de aquí la importancia de búsqueda de estrategias en las diferentes disciplinas de la ciencia de datos e inteligencia artificial. Se pretende dar prioridad a los objetivos de los Programas Nacionales Estratégicos (Pronaces) Agua y Vivienda [3], en este punto se han identificado datos abiertos para ambos pronaces y se establece una estrategia para identificar variables de interés entre ambos conceptos, cuya finalidad será evaluar la relación del consumo de agua y el crecimiento de las ciudades de acuerdo con los

datos recuperados. Se requiere hacer un análisis preliminar del conjunto y el análisis de modelos descriptivos, se establecerán estrategias para analizar el conjunto de datos final y realizar su análisis prescriptivo.

Referencias

1. Breña-Naranjo, A.: Hacia una política nacional de datos abiertos e inclusión digital en el sector hídrico. Perspectivas IMTA, vol. 2, no. 38 (2021) doi: 10.24850/b-imta-perspectivas-2021-38
2. Cámara de Diputados: Constitución Política de los Estados Unidos Mexicanos: Página oficial (2024) www.diputados.gob.mx/LeyesBiblio/ref/cpeum.htm
3. Consejo Nacional de Humanidades, Ciencias y Tecnologías: Programas nacionales estratégicos. Agua y Vivienda (2024) conahcyt.mx/pronaces/
4. Gobierno de México: Política de transparencia, gobierno abierto y datos abiertos: Página oficial (2024)www.gob.mx/sfp/documentos/politica-de-transparencia-gobierno-abierto-y-datos-abiertos?state=published
5. Instituto Nacional de Estadística y Geografía: Demografía y sociedad, vivienda: Página oficial (2024) www.inegi.org.mx
6. OpenET: Página oficial (2024) explore.etdata.org
7. Organización de las Naciones Unidas: Página oficial (2024) www.un.org/es
8. Padilla, R.: Una persona necesita 100 litros de agua al día. Organización Mundial de la Ssalud, Gaceta UDG (2024) www.gaceta.udg.mx/una-persona-necesita-100-litros-de-agua-al-dia-oms/
9. Ramos-Bueno, A.: Análisis socio-espacial del uso doméstico del agua en la Ciudad de México. Tésis de Maestría, Universidad Naiconal Autónoma de México (2016)
10. SASGIS: Imágenes satelitales (2024) www.sasgis.org
11. Secretaría de Gobernación: Diario oficial de la federación: Plan nacional de desarrollo 2019-2024 (2024) www.dof.gob.mx/nota_detalle.php?codigo=5565599&fecha=12/07/2019#gsc.tab=0
12. Townsend, A., Zambrano, P.: Big data urbana: Una guía estratégica para las ciudades. Banco Interamericano para el Desarrollo: Página oficial (2024) publications.iadb.org
13. Vázquez, H., Scetta, A., Piedrafita, C.: URSA una herramienta para guiar la planificación urbana con datos satelitales (2023) blogs.iadb.org/ciudades-sostenibles/es/ursa-una-herramienta-para-guiar-la-planificacion-urbana-con-datos-satelitales

ISSN 1870-4069

Ambiente de realidad virtual para terapia de exposición de la acrofobia

Brian Martín Aguilar-González[1], Juan Carlos González-Islas[2], Gildardo Godinez-Garrido[1, 2], Vanessa Monserrat Vázquez-Vázquez[3], Ma. de Jesus Gutierrez-Sánchez[2]

Universidad Tecnológica de Tulancingo,
Hidalgo,
México

Universidad Autónoma del Estado de Hidalgo,
Hidalgo,
México

Universidad de Guadalajara,
Guadalajara,
México

{1718110564, gildardo.godinez}@utectulancingo.edu.mx
{juan_gonzalez7024, madejesus_gutierrez}@uaeh,edu.mx
{vanessa_vazquez4619}@alumnos.udg.edu.mx

Resumen. La realidad virtual se ha convertido en una herramienta asistencial efectiva en el tratamiento de las fobias. El presente trabajo está enfocado en el desarrollo de un ambiente virtual para su uso como herramienta asistencial de la terapia expositiva de acrofobia. Para el desarrollo y ejecución del ambiente se emplearon el motor gráfico de Unity, software de diseño asistido por computadora (CAD, por sus siglas en inglés), un gamepad y un teléfono inteligente. En esta primera etapa, se evaluó la usabilidad del sistema mediante la escala de usabilidad de un sistema (SUS, por sus siglas en inglés). La prueba fue realizada por 10 voluntarios con miedo a las alturas (5 mujeres y 5 hombres), los cuales recorrieron los escenarios ordenados de acuerdo a su nivel de complejidad. Los 73 puntos obtenidos en la escala SUS, determinaron un resultado aceptable de usabilidad. Se identificaron como áreas de mejora la generación de un mayor número de escenarios y el realismo de estos. Asimismo, en futuros estudios se deberá realizar pruebas de funcionalidad en muestras clínicas con acrofobia.

Palabras clave: Realidad virtual, acrofobia, tratamiento expositivo, CAD, unity.

Virtual Reality Environment for Exposure Therapy of Acrophobia

Abstract. Virtual reality has become an effective healthcare tool in the treatment of phobias. The present work focuses on the development of a virtual environment for use as a care tool for the therapy of exposure to acrophobia. For the development and execution of the environment, the Unity graphics engine, computer-aided design (CAD) software, a gamepad and a smartphone were used.

In this first stage, the usability of the system was evaluated using the system usability scale (SUS). The test was carried out by 10 volunteers with fear of heights (5 women and 5 men), who went through the scenarios ordered according to their level of complexity. The 73 points obtained on the SUS scale determined an acceptable usability result. Areas for improvement were identified for the generation of a greater number of scenarios and their realism. Likewise, in future studies functionality tests should be performed on clinical samples with acrophobia.

Keywords: Virtual reality, acrophobia, exposure therapy, CAD, unity.

1. Introducción

La realidad virtual (RV) permite al usuario interactuar con ambientes simulados por computadora, los cuales representan al mundo real o uno imaginario [4]. Además de los video juegos, la RV se ha aplicado en áreas como la medicina, la industria aeroespacial, la arquitectura, la arqueología, la anatomía, la fisiología, la milicia, la psicoterapia, entre otras [20]. Si bien en la actualidad hay mejoras sustanciales en las tecnologías de RV, éstas requieren ciertas mejoras. El nivel de aceptación con latencia baja, la renderización en tiempo real, la resolución y adaptación para cada pantalla, así como el contacto háptico son áreas de oportunidad. De igual manera, es necesario considerar factores como: la interacción efectiva con el mundo virtual de manera física y sensorial, el modelado y operación eficiente basado en imágenes y mundos inexistentes. En cuanto al factor humano, un aspecto a considerar es el malestar fisiológico a la exposición a esta tecnología, el cual debe ser estudiado y atendido por los especialistas [22, 25].

1.1. Realidad virtual y acrofobia

Una fobia se caracteriza por la presencia de ansiedad incontrolable no coherente al riesgo real y que interfiere con el desenvolvimiento cotidiano de las personas y por ende genera malestar significativo [24]. Aproximadamente un tercio de la población mundial es susceptible a la acrofobia y la intolerancia visual a las alturas [18]. En México se ha estimado que el 7.1 % de las personas padecen alguna fobia, siendo una de las patologías mentales más frecuentes, siendo en las mujeres en quienes se presenta en una mayor proporción en comparación con los varones (7.3 %) [21], por lo que es importante contribuir con soluciones a esta problemática

La acrofobia es una fobia específica caracterizada por ansiedad habitual, inmediata, irracional y la evitación de situaciones que impliquen altura (p.ej., subir a puentes, ascender a través de escaleras, ver a través de un vidrio dentro de un edificio, viajar en avión, etc.) [16]. Existen diferentes tratamientos para las fobias como son las terapias: de exposición, desensibilización sistemática, cognitivas, de relajación y/o farmacoterapia [23]. En lo que atañe a las técnicas de exposición de manera general consisten en exponer gradualmente al paciente a la fuente de ansiedad en un entorno gestionado y controlado, mediante imágenes, vídeos o exposiciones directas [8].

En el caso de la acrofobia, la literatura ha referido con mayor empleo: desensibilización sistemática, técnicas de programación neurolingüística, atención plena y exposición en vivo o a través de RV [7, 13]. Estos tratamientos ayudan a disminuir la sintomatología de la ansiedad ante el estimulo temido [2]. Actualmente, existen múltiples trabajos para el tratamiento expositivo de la acrofobia mediante la RV. Un reciente meta-análisis en red de ensayos controlados identificó que la psicoterapia en la que se contó con un entrenador de realidad virtual tuvo una mayor efectividad en comparación con otros 19 tratamientos, aunque aún es necesaria mayor evidencia que consolide dichos resultados [1, 6].

El uso de la RV dirigido a la exposición de alturas, además de utilizarse en el ámbito clínico como apoyo para el tratamiento de la acrofobia se ha empleado también en deportes como el paracaidismo [17] o en trabajos que deben realizarse en las alturas [5]. Aunado a lo anterior, el avance tecnológico en los teléfonos inteligentes ha permitido su uso para la interacción con sistemas de RV centrados en tratamiento de la acrofobia como lo es ZeroPhobia [11].

Por otra parte, los sensores del teléfono como el acelerómetro y el giroscopio, permiten al usuario la interacción kinestésica con el ambiente [15]. En otros trabajos se han empleado tecnologías de propósito específico como es el Oculus Rift [3, 9], o basados en el visor Google Cardboard [19]. Derivado de lo anterior, este proyecto se centró en el desarrollo de un marco de trabajo para el tratamiento expositivo de acrofobia empleando RV. El cual propone la medición de datos fisiológicos y la evaluación de la usabilidad como una primera etapa, mediante una prueba experimental y la metodología SUS.

2. Marco de trabajo para el tratamiento de acrofobia empleando realidad virtual

Este marco de trabajo fue proyectado y desarrollado de manera integral por un equipo interdisciplinario integrado por especialistas del área de psicología y de RV (Figura 1). Los elementos principales del sistema de RV son las plataformas de desarrollo, el sistema de interacción; y el sistema de prueba y evaluación. El desarrollo del ambiente virtual se realizó empleando el motor gráfico de Unity. En dicho motor se incorporaron los modelos 3D previamente desarrollados en software de CAD, las físicas de los objetos, la programación del ambiente virtual y la exportación de los archivos para Android.

El sistema de interacción permite al usuario interactuar de manera perceptiva con el mundo virtual, mediante auriculares con Bluetooth, un teléfono inteligente con sistema operativo Android®, gafas para realidad virtual, y un gamepad genérico. El teléfono para ejecución y visualización, tiene un procesador Qualcomm®Snapdragon™665, iGPU Adreno™610, 4 GB de RAM, 64 GB de ROM, pantalla de 6 pulgadas, y un giroscopio. Los requerimientos mínimos para correr la aplicación son: sistema operativo Android 5.1, Procesador Snapdragon™460 o Mediatek™Helio P60, iGPU Adreno™540 o Mali™G50, Memoria RAM de 3 GB, almacenamiento 70 MB de descarga y 250 MB instalado y un tamaño de pantalla mínimo de 5.5 pulgadas.

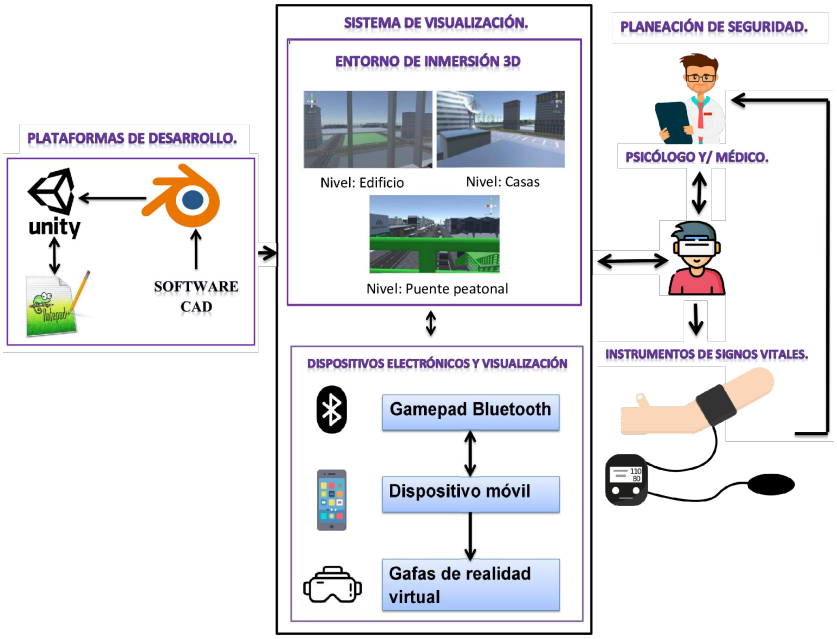


Fig. 1. Marco de trabajo para el tratamiento de acrofobia empleando RV.

El uso de la RV como herramienta para el tratamiento de exposición en la requiere un ambiente controlado, el cual puede apoyarse (Figura 1) en la adquisición y análisis de datos fisiológicos del usuario durante la terapia; así como el acompañamiento de un especialista clínico. Los signos vitales más relevantes y que suben sus niveles cuando una persona sufre de fobia son la frecuencia cardiaca, ritmo respiratorio los cuales, al ser sobrepasados, son un indicador emergente de criterio de paro de la sesión. El diagrama de flujo de la Figura 2 describe la secuenciación de las animaciones dependiendo las condiciones programadas. Existen 3 condiciones, los cuales son: 1. Punto de observación o Raycaster ejecutados, 2. Objeto observado interactivo y 3.

Ejecución de acción mediante botón activado. Con estas tres condiciones, se puede realizar la ejecución de pausa o reproducción de animaciones en distintos objetos que tienen animado un archivo de animación. Uno de los componentes principales en un sistema de RV es el grado de inmersión y realismo, por lo que mediante el uso de sensores inerciales como el giroscopio es posible determinar el movimiento angular de la cabeza y controlar la escena como en el sistema de visión natural (Figura 3).

2.1. Escenas

Con el propósito de proporcionar una experiencia más inmersiva para el usuario, el entorno virtual desarrollado en este trabajo contiene tres escenarios comunes en las ciudades en México. El usuario tiene restringida la exploración libre, es decir, únicamente puede explorar los escenarios asignados por los desarrolladores con base en las recomendaciones de la especialista.

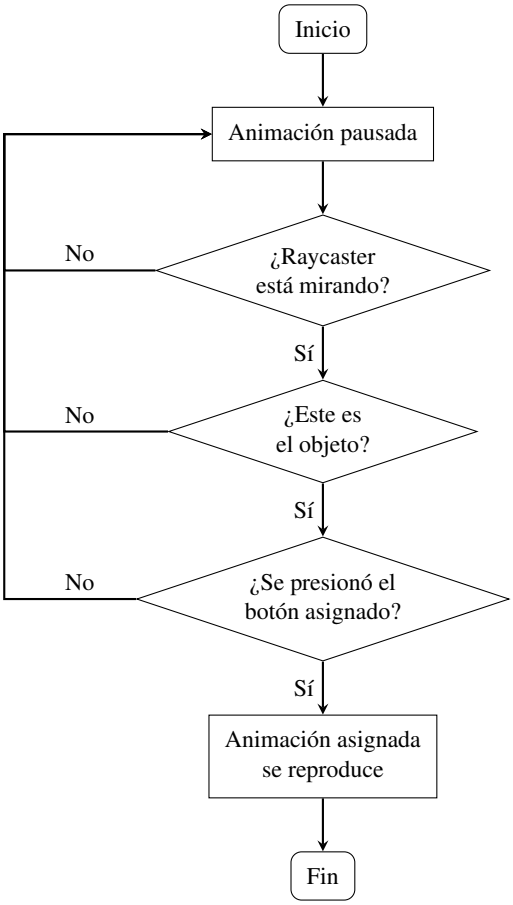


Fig. 2. Diagrama de flujo para secuenciación de animaciones.

Los tres escenarios son: un puente peatonal, el techo de una casa y un edificio con un elevador trasparente. Todos los objetos de cada uno de los ambientes fueron primero diseñados y modelados empleando software de CAD.

2.2. Escena del puente

El escenario del puente es el menos riesgoso en ´términos de acrofobia para el usuario. En México, la mayoría de ciudades cuentan con puentes peatonales, por lo que es cruzar por éstos es un escena cotidiana. Este escenario se modeló con base en un puente peatonal real, en el que la altura promedio es de 5 metros con referencia al piso. En la Figura 4 se muestra una vista panorámica desde arriba del puente en el ambiente de Unity. Este escenario cuenta con 3 objetos que actúan como portales para llevar al usuario al menú de selección de niveles, los cuales se encuentran en la primera sección de escaleras (Figura 5). Además, se incorporan animaciones de vehículos circulando debajo del puente como se ve en la Figura 6.

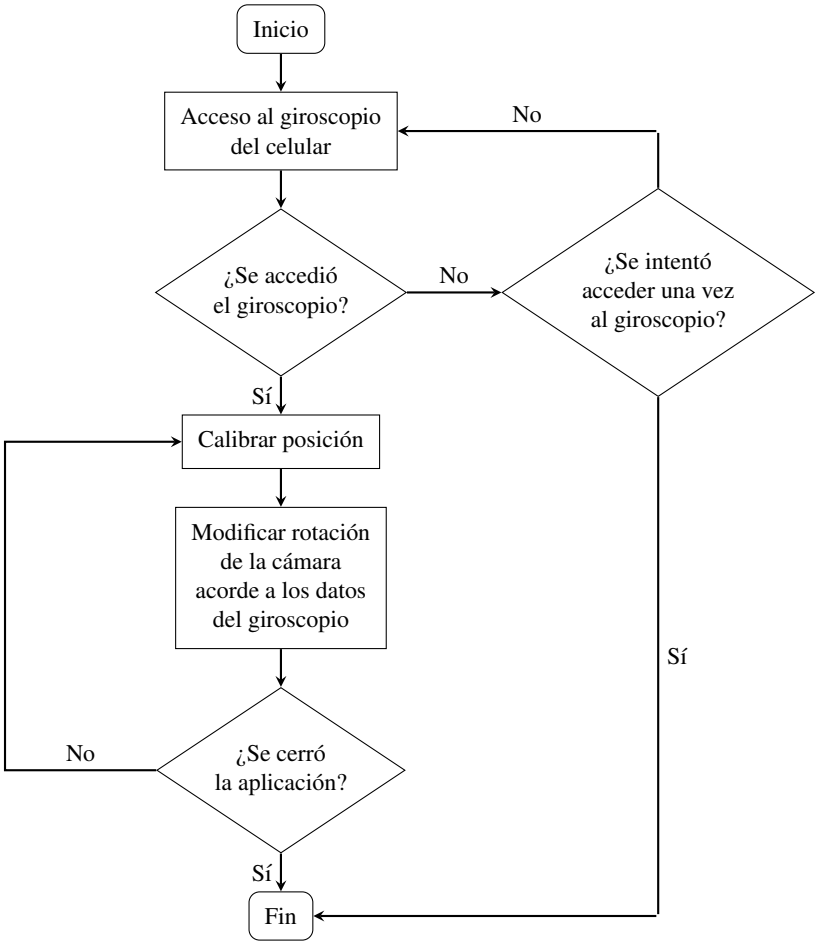


Fig. 3. Diagrama de flujo de control de escena mediante el giroscopio.

La emulación de los estilos urbanos de cada país o región del mundo es muy complicada debido a la diversidad de modelos existentes, por lo que cada ambiente se diseña con un propósito específico, conservando la idea general.

2.3. Escena de la casa

El segundo escenario, con un nivel intermedio de riesgo, es el techo de una casa. Es común que las personas realicen actividades de limpieza o reparación en los techos de su casa, por lo que es un escenario propicio para simular este ambiente para las personas con acrofobia. En esta escena, el reto implica subir una escalera hasta el techo, para luego manipular una antena satelital. Asimismo, el usuario deberá bajar y tomar el objeto que simula el portal para regresar al menú de selección de niveles. En la Figura 7 se presenta una vista en perspectiva desde el ambiente de Unity de la escena del techo de la casa.

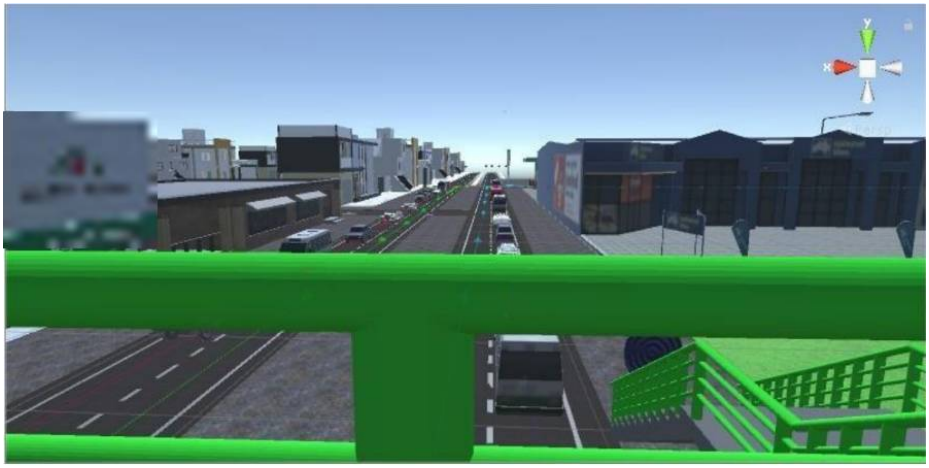


Fig. 4. Escena del puente desde el entorno virtual en Unity.

2.4. Escena del edificio

Finalmente, el ambiente más riesgoso es el de un edificio y un elevador transparente, ya que son en los que se percibe una mayor altura. En la escena del edificio los usuarios experimentan la aproximación a diferentes alturas en un edificio de 5 pisos, en los cuales del piso 2 al 5 se muestra la altura. En estos pisos, existe una animación de ambiente de oficina mediante una computadora funcionando y desplegando código en pantalla. El usuario puede explorar el piso o salir utilizando un objeto en la pared. En la Figura 8 se muestra una vista en perspectiva desde el ambiente de Unity hacia a fuera del Edificio. En este caso, esta escena representa el nivel con la dificultad mayor, con base en la altura. La Figura 9 presenta la vista hacia afuera del edficio desde la perspectiva del usuario.

3. Evaluación de usabilidad

Para evaluar la usabilidad del sistema, se seleccionaron 10 participantes que refirieron tener cierto grado de temor a las alturas. Quienes hicieran el recorrido por los tres escenarios. Cabe mencionar que durante la prueba de usabilidad se mantuvieron presentes el desarrollador del sistema como la psicóloga que asesoró el proyecto. El funcionamiento del sistema, ejecutado por usuario consiste en:

1. Instalar la aplicación en un teléfono inteligente con sistema operativo Android.
2. Ejecutar la aplicación en el teléfono inteligente.
3. Colocar las gafas de realidad virtual que contienen el teléfono con la aplicación instalada.
4. Hacer el recorrido por los escenarios asignados de acuerdo a su nivel de fobia.

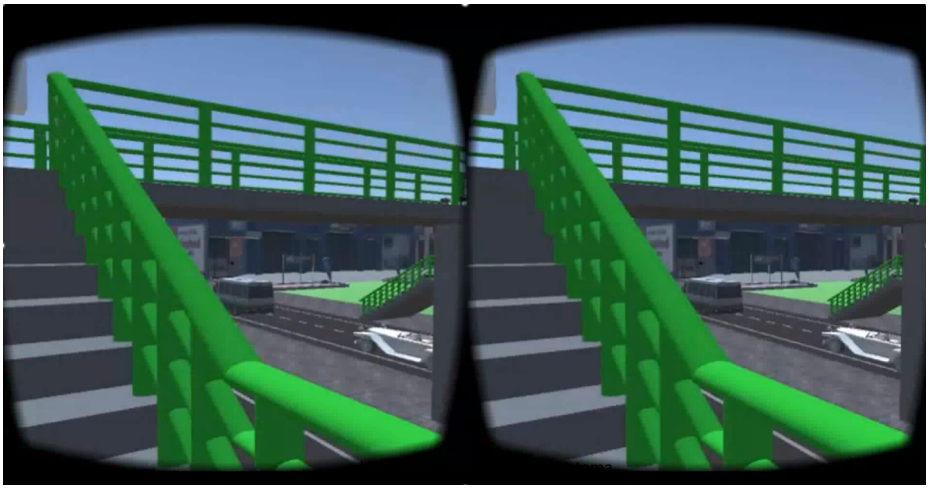


Fig. 5. Escena para subir el puente desde la perspectiva del usuario.

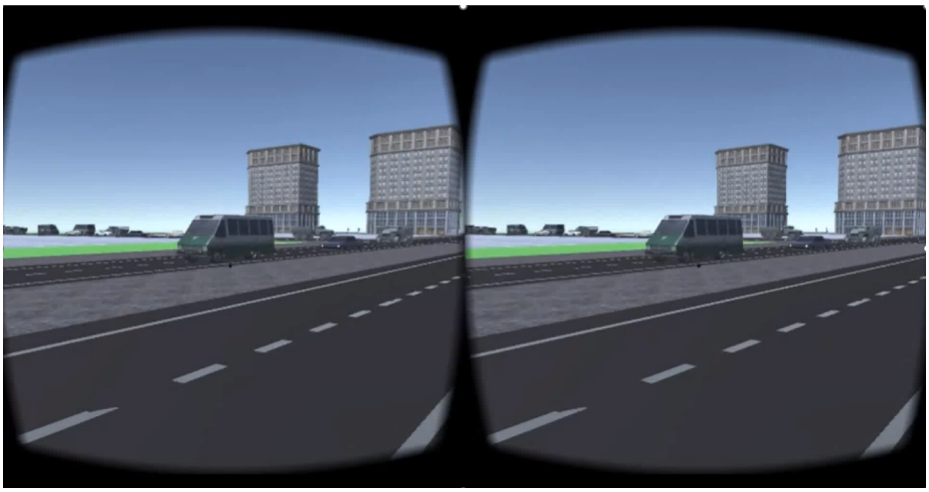


Fig. 6. Vista panorámica desde el puente desde la perspectiva del usuario.

Durante la sesión se debe monitorear el nivel de estrés del paciente a través de la medición de la temperatura corporal y el ritmo cardiaco para determinar la progresión con la que debería continuarse el recorrido. Una vez realizado el recorrido virtual por los 3 escenarios, los usuarios contestaron una encuesta con las preguntas de la metodología de la escala de usabilidad del sistema (SUS) [12]. Una prueba heurística representa un mayor costo de prueba debido a la experiencia requerida por los evaluadores, incluso cuando significa un proceso de prueba más fácil. Aunque las pruebas y evaluaciones del SUS pueden presentar un procedimiento más complicado en el manejo de información, requiere una pequeña cantidad de muestras obteniendo buenos resultados [10].

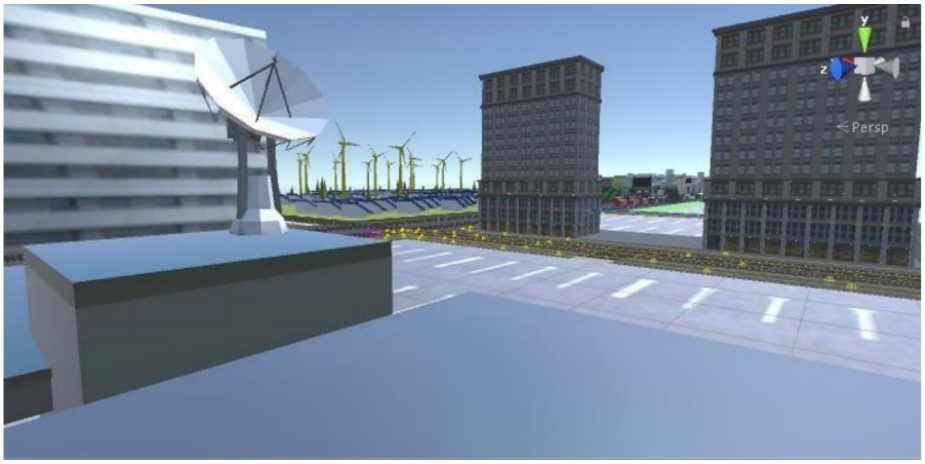


Fig. 7. Escena del techo de una casa desde el entorno virtual en Unity.

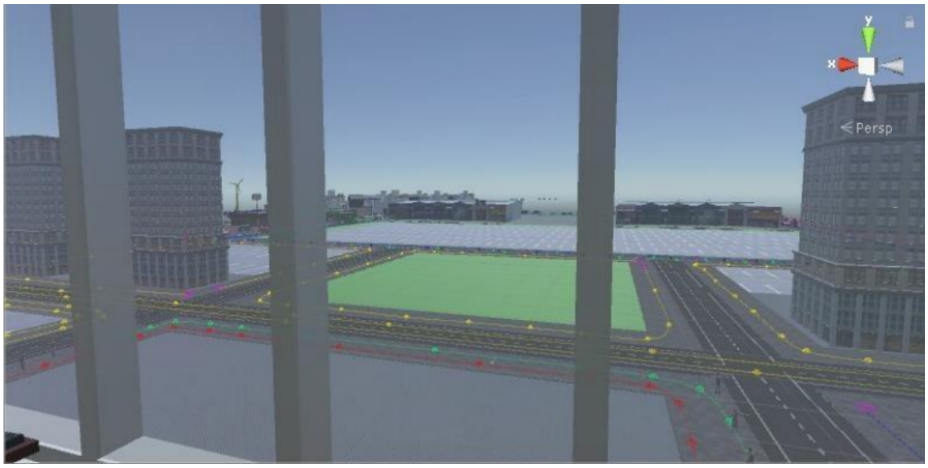


Fig. 8. Escena del edificio desde el entorno virtual en Unity.

Para estudios de usabilidad en fases iniciales como el propuesto en este trabajo, con 5 evaluaciones es suficiente. Las preguntas que integran la encuesta de usabilidad son.

1. Creo que me gustaría utilizar este sistema con frecuencia.
2. El sistema me pareció innecesariamente complejo.
3. Pienso que el sistema era fácil de usar.
4. Creo que necesitaría el apoyo de un técnico para poder utilizar este sistema.
5. Encontré que las diversas funciones de este sistema están bien integradas.
6. Pensé que había demasiada inconsistencia en este sistema.

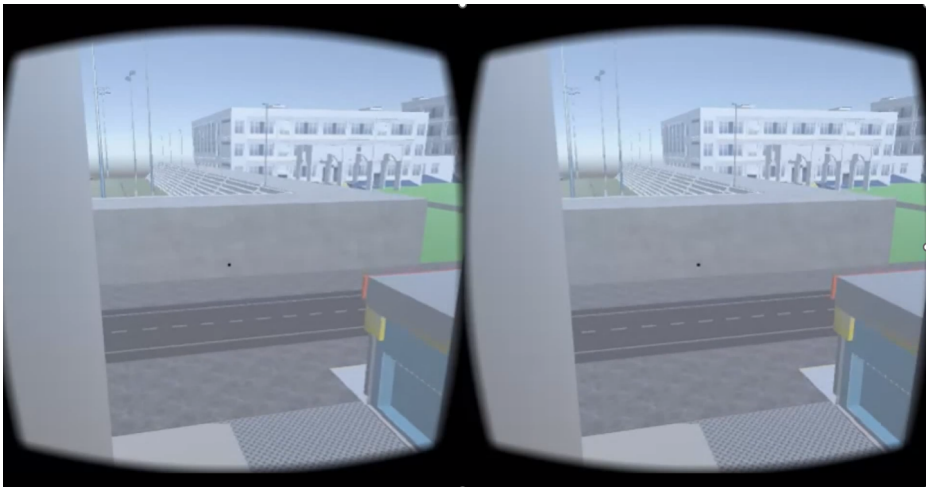


Fig. 9. Vista hacia afuera del edificio desde la perspectiva del usuario.

7. Me imagino que la mayoría de la gente aprendería a utilizar este sistema muy rápidamente.
8. El sistema me parece muy complicado de utilizar.
9. Me siento muy seguro al usar el sistema.
10. Necesito aprender muchas cosas antes de poder empezar a utilizar este sistema.

De los 10 enunciados, los 5 impares corresponden a respuestas positivas, mientras que los 5 pares, se refieren a percepciones negativas. Las opciones para cada respuesta son las siguientes:

1. Totalmente en desacuerdo.
2. En desacuerdo.
3. Neutral.
4. De acuerdo.
5. Totalmente de acuerdo.

La calificación de la escala de usabilidad considera que el mismo valor ordinal de la respuesta, es el valor de puntaje que se le asigna. Para cada una de las preguntas impares, se resta 1 de la puntuación. Para cada una de las preguntas pares, se resta su valor a 5. Posteriormente, se suman esos nuevos valores calculados y se multiplican por 2.5 para tener una escala con base 100. Se puede observar que las preguntas evalúan la percepción subjetiva de los usuarios de forma objetiva mediante el uso de una escala Likert. Esta escala evaluará luego el nivel de acuerdo o desacuerdo del usuario en sus dimensiones cognitiva y conductual [14].

Tabla 1. Porcentajes de las respuestas de los usuarios durante la evaluación SUS por cada una de las respuestas.

Pregunta	Totalmente en Desacuerdo	En Desacuerdo	Neutro	De Acuerdo	Totalmente de Acuerdo
1	0	0	30	40	30
2	30	30	40	0	0
3	0	0	50	30	20
4	30	50	20	0	0
5	0	10	30	30	30
6	20	20	60	0	0
7	0	0	10	70	20
8	10	50	30	10	0
9	0	0	0	40	60
10	30	40	30	0	0

4. Discusión de resultados

La evaluación de usabilidad del sistema fue realizada por 10 participantes que refirieron cierto temor a las alturas (5 hombres y 5 mujeres). Los usuarios tuvieron una edad entre los 18 y 50 años, pertenencientes a la ciudad de Tulancingo, Hidalgo, México. La mayoría de los voluntarios señalaron no tener experiencia previa en juegos usando RV. 5 de los participantes lograron recorrer los tres escenarios sin asistencia adicional al tutorial inicial, sin embargo, durante el recorrido hubo algunas pausas para inmergir en la altura de la escena. 3 de los participantes requirieron asistencia del desarrollador, tanto para elegir el escenario como para regresar al menú de opciones, Por último, 2 de ellos, no recorrieron los 3 escenarios debido a su falta de experiencia con la RV y el miedo a las alturas.

La escena que lograron recorrer fue la del puente, lo que otorga una perspectiva del sistema en general. En general, los participantes informaron a nuestro equipo que el ambiente era atractivo, interactivo, y desafiante. Algunos comentarios para mejorar el ambiente virtual, se centraron en: agregar nuevos escenarios, mejorar los gráficos y colores, clarificar las instrucciones sobre el recorrido, así como proveer de más tiempo para familiarizarse con el sistema y que el sistema sea más intuitivo para la elección de la escena.

La especialista que apoyó desde el área de la psicología (VMVV) subrayó la utilidad que esta herramienta de RV podría tener en el entorno clínico, para lo cual será importante continuar trabajando con las áreas de oportunidad identificadas en el estudio como la evaluación de la herramienta en muestras clínicas (pacientes diagnosticados con acrofobia) en tanto a la usabilidad como en su efecto en la sintomatología de la fobia. Por otra parte, se destacó la importancia de la interacción con diferentes niveles en el entorno virtual pudiendo implicar diferentes objetos y tareas, la facilidad del modo de movimiento en el recorrido. Además, incorporar un tutorial para el uso de la herramienta dirigido a los potenciales usuarios; profesionales y pacientes. De igual manera, durante la sesión, se realizó la medición de la temperatura y ritmo cardiaco en los usuarios, con el objetivo de monitorear su estado fisiológico y abortar el recorrido ante cualquier anomalía fisiológica.

El ritmo cardiaco de los usuarios durante el recorrido virtual fue de 67 pulsaciones por minuto, lo cual se considera normal; sin embargo, algunos de los usuarios experimentaron un máximo de 85 pulsaciones por minuto en las zonas más altas en el ambiente virtual. En cuanto a la temperatura corporal, el promedio fue de 36.2 grados, la cual se mantuvo en ese promedio.

Otro, elemento importante es el teléfono usado, los equipos con capacidades menores a las recomendadas experimentaron lentitud en la ejecución, tiempo de carga largos, caída de FPS en los espacios con más elementos 3D y en casos menos frecuentes, cierre de la aplicación por desbordamiento de memoria RAM, poca potencia gráfica o sobrecalentamiento.

En términos de usabilidad del ambiente virtual, derivado de las encuestas realizadas a los participantes, la puntuación de la escala de usabilidad del sistema propuesto fue de 73, mayor a la puntuación promedio. Si bien, la escala no es un diagnóstico y no identifica los problemas específicos, si se tiene una referencia para conocer los parámetros para mejorar la usabilidad. En la Tabla 1 se muestra de manera resumida, los valores del porcentaje de las 10 evaluaciones para cada una de las 10 preguntas.

5. Conclusiones

El ambiente virtual desarrollado es congruente con otros antecedentes realizados para el tratamiento de esta patología, aunque tiene como una de sus limitantes no haber sido probado en una muestra clínica. El ambiente permite una aproximación sucesiva a situaciones cotidianas del mundo real para personas con acrofobia. Como fortalezas del sistema desarrollado es que los recursos tecnológicos empleados son mínimos, escalables y adaptables para tratamiento de otro tipo de fobias. El uso de los sensores embebidos en el teléfono inteligente, los sensores vestibles en el usuario, así como la supervisión de los especialistas técnico y clínico, proveen la retroalimentación eficiente para determinar el criterio de paro del recorrido.

De igual manera, el recorrido se puede manejar por niveles adaptándose a las necesidades del usuario, a partir de la sintomatología de la acrofobia que se tenga. La incorporación de sonidos, terapéuticamente ayuda a los usuarios a mejorar el aprovechamiento del interfaz Finalmente, como trabajo futuro se plantea la incorporación de un sistema de monitoreo automático de variables fisiológicas, el desarrollo de más escenarios, la mejora de la renderización, procesamiento de imágenes e intuitividad del sistema, así como la evaluación del ambiente con una muestra clínica estadísticamente significativa.

Referencias

1. Abdullah, M., Shaikh, Z. A.: An effective virtual reality based remedy for acrophobia. International Journal of Advanced Computer Science and Applications, vol. 9, no. 6 (2018) doi: 10.14569/ijacsa.2018.090623
2. Anton, C., Mitruţ, O., Moldoveanu, A., Moldoveanu, F., Kosinka, J.: A serious VR game for acrophobia therapy in an urban environment. In: IEEE International Conference on Artificial Intelligence and Virtual Reality, pp. 258–265 (2020) doi: 10.1109/aivr50618.2020.00054

3. Arroll, B., Wallace, H. B., Mount, V., Humm, S. P., Kingsford, D. W.: A systematic review and meta-analysis of treatments for acrophobia. Medical Journal of Australia, vol. 206, no. 6, pp. 263–267 (2017) doi: 10.5694/mja16.00540
4. Botella, C., Fernández-Álvarez, J., Guillén, V., García-Palacios, A., Baños, R.: Recent progress in virtual reality exposure therapy for phobias: A systematic review. Current Psychiatry Reports, vol. 19, no. 7, pp. 1–13 (2017) doi: 10.1007/s11920-017-0788-4
5. Chardonnet, J. R., Di-Loreto, C., Ryard, J., Housseau, A.: A virtual reality simulator to detect acrophobia in work-at-height situations. In: IEEE Conference on Virtual Reality and 3D User Interfaces, pp. 747–748 (2018) doi: 10.1109/VR.2018.8446395
6. Chou, P. H., Tseng, P. T., Wu, Y. C., Chang, J. P. C., Tu, Y. K., Stubbs, B., Carvalho, A. F., Lin, P. Y., Chen, Y. W., Su, K. P.: Efficacy and acceptability of different interventions for acrophobia: a network meta-analysis of randomised controlled trials. Journal of affective disorders, vol. 282, pp. 786–794 (2021) doi: 10.1016/j.jad.2020.12.172
7. Coelho, C. M., Waters, A. M., Hine, T. J., Wallis, G.: The use of virtual reality in acrophobia research and treatment. Journal of Anxiety Disorders, vol. 23, no. 5, pp. 563–574 (2009) doi: 10.1016/j.janxdis.2009.01.014
8. Darooei, R., Vard, A., Rabbani, H.: BarBam: A new arcophobia virtual reality game. In: International Serious Games Symposium, vol. 1, pp. 48–53 (2019) doi: 10.1109/isgs49501.2019.9047012
9. de-Oliveira, R. E. M., de-Oliveira, J. C.: Virtual environments for the treatment of acrophobia ambientes virtuais para tratamento de acrofobia. In: 20th Symposium on Virtual and Augmented Reality, pp. 37–46 (2018) doi: 10.1109/svr.2018.00018
10. Diazgiron-Aguilar, D., Gonzalez-Islas, J. C., Godinez-Garrido, G., Guzman-Alvarado, M.: Virtual lab environment for programmable logic controllers training. In: XXIV Robotics Mexican Congress, IEEE, vol. 13, pp. 60–65 (2022) doi: 10.1109/comrob57154.2022.9962262
11. Donker, T., Cornelisz, I., van-Klaveren, C., van-Straten, A., Carlbring, P., Cuijpers, P., van-Gelder, J. L.: Effectiveness of self-guided app-based virtual reality cognitive behavior therapy for acrophobia: A randomized clinical trial. JAMA Psychiatry, vol. 76, no. 7, pp. 682 (2019) doi: 10.1001/jamapsychiatry.2019.0219
12. Ependi, U., Kurniawan, T. B., Panjaitan, F.: System usability scale vs heuristic evaluation: A review. Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer, vol. 10, no. 1, pp. 65–74 (2019) doi: 10.24176/simet.v10i1.2725
13. Giraldy, D. J., Novaldo, W.: A systematic literature review: Acrophobia treatment with virtual reality. Engineering, Mathematics and Computer Science Journal, vol. 4, no. 1, pp. 33–38 (2022) doi: 10.21512/emacsjournal.v4i1.8077
14. Joshi, A., Kale, S., Chandel, S., Pal, D.: Likert scale: Explored and explained. British Journal of Applied Science and Technology, vol. 7, no. 4, pp. 396–403 (2015) doi: 10.9734/bjast/2015/14975
15. Lengkong, O., Bororing, J.: Hyperion: A simulation of high places in the form of virtual reality for acrophobia sufferers. In: 2nd International Conference on Cybernetics and Intelligent System, pp. 1–4 (2020) doi: 10.1109/icoris50180.2020.9320824
16. Morrison, J.: DSM-5® Guía para el diagnóstico clínico. Editorial El Manual Moderno (2015)
17. Navas-Moya, M. P., Mayorga-Soria, P., Mayorga-Soria, P., Navas-Moya, M.: Aplicación de realidad virtual en el tratamiento de acrofobia en personas que practican deportes extremos como el paracaidismo. Informática y Sistemas: Revista de Tecnologías de la Informática y las Comunicaciones, vol. 4, no. 1, pp. 1–12 (2020) doi: 10.33936/isrtic.v4i1.2283
18. Rimer, E., Husby, L. V., Solem, S.: Virtual reality exposure therapy for fear of heights: Clinicians' attitudes become more positive after trying VRET. Frontiers in Psychology, vol. 12 (2021) doi: 10.3389/fpsyg.2021.671871

19. Rique-Gambini, J. A.: Sistema de RV google cardboard para mejorar el nivel de la acrofobia de los pacientes de psicosalud equilibrio (2020)
20. Rowland, D. P., Casey, L. M., Ganapathy, A., Cassimatis, M., Clough, B. A.: A decade in review: A systematic review of virtual reality interventions for emotional disorders. Psychosocial Intervention, vol. 31, no. 1, pp. 1–20 (2021) doi: 10.5093/pi2021a8
21. Secretaría de Salud: Fobias, trastornos mentales más comunes (2017) www.gob.mx/salud/prensa/073-fobias-trastornos-mentales-mas-comunes
22. Shanmugam, M., Sudha, M., Lavitha, K., Venkatesan, V. P., Keerthana, R.: Research opportunities on virtual reality and augmented reality: A survey. In: IEEE International Conference on System, Computation, Automation and Networking, pp. 1–6 (2019) 10.1109/ICSCAN.2019.8878796
23. Silva-Freitas, J. R., Silva-Velosa, V. H., Nunes-Abreu, L. T., Lucas-Jardim, R., Vieira-Santos, J. A., Peres, B., Campos, P. F.: Virtual reality exposure treatment in phobias: A systematic review. Psychiatric Quarterly, vol. 92, no. 4, pp. 1685–1710 (2021) doi: 10.1007/s11126-021-09935-6
24. Weder, N. D., Aziz, R., Wilkins, K., Tampi, R. R.: Frontotemporal dementias: A review. Annals of general psychiatry, vol. 6, no. 1, pp. 1–10 (2007) doi: 10.1186/1744-859x-6-15
25. Zhang, Y., Liu, H., Kang, S. C., Al-Hussein, M.: Virtual reality applications for the built environment: Research trends and opportunities. Automation in Construction, vol. 118, pp. 103311 (2020)

ISSN 1870-4069

Uso de SVM en señales EEG para la clasificación de comandos mentales y su aplicación para el control de dispositivos móviles

Vanessa Isabel Arellano-Serna, Aurora Torres-Soto,
María Dolores Torres-Soto

Universidad Autónoma de Aguascalientes,
Ciudad Universitaria,
México

{al227437, aurora.torres, dolores.torres}@edu.uaa.mx

Resumen. En el siglo XX surgieron dos elementos fundamentales: las máquinas de soporte vectorial (SVM) y las interfaces cerebro-computador (BCI). Las SVM, algoritmos de aprendizaje supervisado, han evolucionado para resolver problemas no lineales, mientras que las BCI buscan la comunicación entre cerebro y computadora. Este estudio propone una solución que integra SVM para clasificar comandos mentales en señales electroencefalográficas (EEG) para el control de dispositivos móviles. Se utilizó la diadema Emotiv Epoc+ de 14 canales para la captura de datos EEG, seguida de la extracción de características mediante el método de patrón espacial común (CSP). Los resultados muestran altas precisiones de clasificación utilizando SVM con kernel RBF, los cuales se encuentran en un promedio de 0.94, 0.95 y 0.95 en las métricas de precisión, recobro y F1 respectivamente. Se propone una definición de comandos mentales y se discute su relevancia en el contexto de las BCI. La arquitectura del sistema propuesto incluye Subapase, Heroku, ThreeJS y Flutter. Este estudio promueve la inclusividad eliminando el esfuerzo físico-motor para la interacción con la tecnología, y representa un paso hacia un cambio de paradigma en la interacción con el mundo digital.

Palabras clave: Señales EEG, SVM, comandos mentales, dispositivos móviles.

Use of SVM on EEG Signals for Mental Command Classification and its Application for Mobile Device Control

Abstract. The 20th century saw the emergence of two fundamental elements: support vector machines (SVM) and brain-computer interfaces (BCI). SVM, which are supervised learning algorithms, have evolved to solve nonlinear problems, while BCI seek brain-computer communication. This study proposes a solution that integrates SVM to classify mental commands found in electroencephalographic (EEG) signals for mobile device control. The 14-channel Emotiv Epoc+ headset was used for EEG data capture, followed by feature extraction using the common spatial pattern (CSP) method. The results show high classification accuracies using SVM with RBF kernel, which

average 0.94, 0.95 and 0.95 on precision, recall and F1 metrics respectively. A definition of mental commands is proposed and their relevance in the context of BCI is also discussed. The proposed system architecture includes Subapase, Heroku, ThreeJS and Flutter. This study promotes inclusivity by eliminating physical-motor required in daily technology usage, and represents a step towards a paradigm shift on how to interact with the digital world.

Keywords: EEG signals, SVM, mental commands, device control.

1. Introducción

A partir del siglo XX surgieron dos elementos fundamentales para el presente trabajo: las máquinas de soporte vectorial (SVM) y las interfaces cerebro computador (BCI). Las primeras surgen como un algoritmo de aprendizaje supervisado utilizado en una gran variedad de problemas de clasificación y regresión. Su objetivo es maximizar el hiperplano que separa a las clases que se desean clasificar y aunque su funcionamiento fue pensado para problemas lineales, este algoritmo ha evolucionado y se apoya de diferentes funciones de kernel para transformar sus características, haciendo así a la SVM más flexible y funcional para problemas no lineales [10, 11].

Por otro lado, BCI es una tecnología que busca la comunicación, como su nombre lo señala, entre cerebro y computadora. Este proceso de intercambio de información puede ser unidireccional (cerebro a computador o computador a cerebro), o bidireccional (cerebro a computador y computador a cerebro) [13]. Uno de los principales métodos de captura de datos para esta tecnología, es la captura de señales electroencefalográficas, una técnica no invasiva para el registro de la actividad eléctrica del cerebro. Estas señales pueden adquirir luego una interpretación, por ejemplo, se pueden convertir en información acerca de si el usuario está relajado, en hiperfoco o distraído.

Sin embargo, las inferencias no se limitan al estado mental en el que se encuentra el usuario; esta tecnología también se ha utilizado para captar la intención de movimiento del usuario y su aplicación se ha enfocado principalmente en trabajos biomédicos [13]. Ahora mismo el campo de los sistemas BCI se encuentra en una etapa crucial, ya que el área ha resuelto situaciones cotidianas, una óptica a la que pertenece este trabajo y dentro de la cual los algoritmos de aprendizaje máquina han protagonizado como herramientas para la clasificación de intenciones dentro de las señales EEG.

Aún se enfrentan desafíos, como la entropía de las señales de un mismo sujeto dependiendo del tiempo y espacio en que se encuentre, sin embargo, las máquinas de soporte vectorial (SVM) se han popularizado dentro de la clasificación de estas señales debido a que son capaces de manejar datos de alta dimensionalidad y han presentado precisiones sobresalientes respecto de otros enfoques en problemas multiclase y manejo de entropía [16, 6]. Este estudio tiene como objetivo la búsqueda de promover una mayor y más amplia inclusión eliminando la necesidad de un esfuerzo físico-motor para la interacción con la tecnología, al basarse únicamente en comandos mentales del usuario. Además se busca formar parte de los primeros pasos hacia el cambio de paradigma en la forma en que se interactúa con el mundo digital.

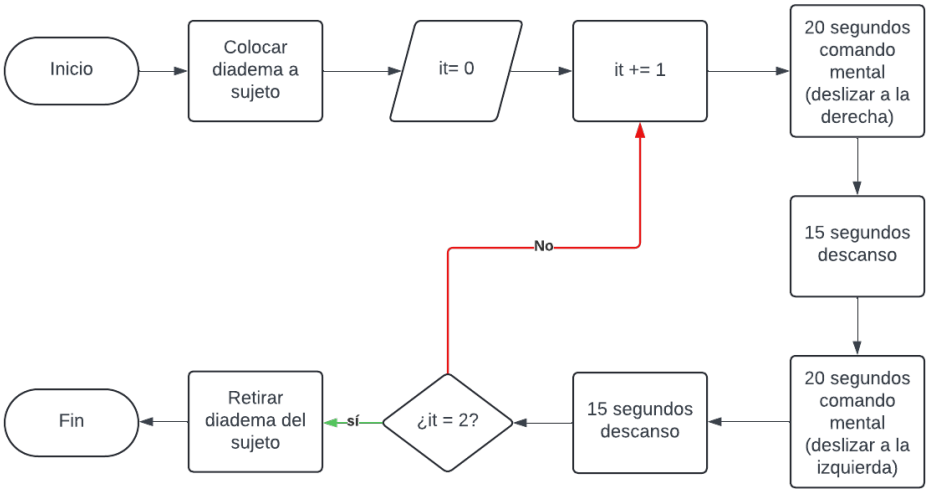


Fig. 1. Diagrama explicativo del proceso de captura de datos.

Se propone una solución que integra el uso de las máquinas de soporte vectorial aplicadas para la clasificación de comandos mentales obtenidos de señales EEG para el control de dispositivos móviles. Para la adquisición de datos EEG se utiliza la diadema Emotiv Epoc+ de 14 canales, seguida de la limpieza de los datos y extracción de características por medio del método CSP (Common Spatial Pattern). Para la demostración visual del funcionamiento de la clasificación de señales EEG, se desarrolló un producto con interfaz gráfica para dispositivos móviles, controlado de manera asíncrona por medio de comandos mentales, para esto se generó una API en Python la cual se integró con una aplicación en Flutter.

En la sección 2. Conceptos principales, se presentan las definiciones más importantes para el entendimiento del trabajo. Seguido por 3. Metodología, donde se describe desde los materiales requeridos, la selección de la población de estudio, el proceso de captura de datos, extracción de características y clasificación. Los hiperparámetros utilizados, resultados y diseño de la arquitectura del sistema pueden ser encontrados en la sección 4. Resultados. Por último en 5. Conclusiones y discusión se destaca la posible relación del método de extracción de características utilizado con el buen desempeño del modelo.

2. Conceptos principales

Esta sección presenta brevemente los conceptos principales utilizados dentro de este trabajo.

1. **SVM**. Tipo de algoritmo de aprendizaje supervisado utilizado principalmente para clasificación y regresión. El objetivo de las SVM es encontrar el hiperplano óptimo que maximice la separación entre las clases en un espacio multidimensional, donde cada punto de datos se representa como un vector [8].

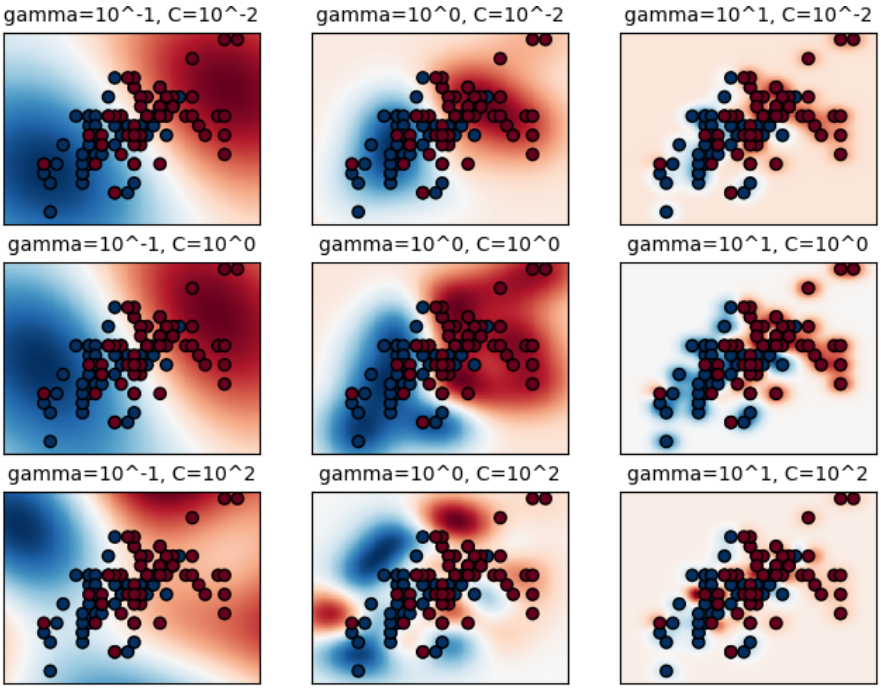


Fig. 2. Representación del impacto de distintos valores de C y γ como mapa de calor [15].

2. **CSP**. Método de extracción de características comúnmente utilizado para procesamiento de señales de clasificación motora, busca de amplificar la variabilidad de los datos entre clases y minimizarla en la misma clase [9].
3. **RBF**. Función kernel que mide la similitud entre pares de puntos de datos basándose en su distancia en el espacio de características [7].
4. **API**. Mecanismo que permite a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos [1].

3. Metodología

3.1. Materiales

Para la obtención de señales EEG, se utiliza la diadema multicanal Emotiv Epoc+, creada para la investigación del cerebro humano y su aplicación en sistemas BCI; cuenta con catorce nodos que permiten una lectura de una amplia área cerebral. Las almohadillas de la diadema deben ser saturadas con solución salina para lograr una buena conexión y recepción de los datos. El dispositivo es acompañado con Emotiv Launcher, un software del fabricante EMOTIV, que permite visualizar la calidad de las señales contacto que se está obteniendo de la diadema con el cuero cabelludo del usuario.

Tabla 1. Hiperparámetros utilizados en la SVM y resultados obtenidos por clase.

Clase	Kernel	Gamma	C	Precisión	recobro	F1
0 (izquierda)	RBF	0.15	2	0.96	0.93	0.95
1 (descanso)				0.94	0.94	0.94
2 (derecha)				0.95	0.98	0.96

3.2. Población de estudio

Para la selección de la población de estudio, se establecieron diversos criterios de inclusión y exclusión, con el fin de controlar en la medida de lo posible las variables que podrían impactar en la entropía de las señales capturadas. Estos criterios son detallados a continuación:

1. **Género**. Con el fin de impulsar la participación de las mujeres en los proyectos del área STEM y buscando disminuir el sesgo de género histórico en la elección de sujetos de prueba, fueron seleccionadas únicamente sujetos de prueba del sexo femenino.
2. **Ausencia de evento cerebrovascular**. Se excluyeron sujetos que hubieran experimentado algún accidente o enfermedad cerebrovascular.
3. **Edad**. Se seleccionaron sujetos de entre 18 y 30 años de edad.
4. **Compromiso de asistencia**. Se requirió que los sujetos de prueba pudieran comprometerse a asistir a todas las sesiones de captura de datos y pruebas, garantizando así la integridad y la coherencia de los datos recopilados a lo largo del estudio.

La selección de sujetos se realizó mediante un proceso de reclutamiento activo, que incluyó la difusión de información sobre el estudio en redes sociales y sociedades de alumnos de la institución. A los individuos interesados se les envió un formulario inicial para determinar su elegibilidad de acuerdo con los criterios mencionados anteriormente, Finalmente, como primera aproximación se seleccionaron dos mujeres de 19 y 22 años, además se obtuvo el consentimiento informado de ambas participantes antes de su inclusión en el estudio.

3.3. Descripción de la captura de datos

Los datos capturables se enfocaron en la intención de dos comandos mentales principales: 1) mover un objeto a la izquierda y 2) mover un objeto a la derecha, entre los comandos, se encuentra un tiempo de descanso que es capturado y etiquetado dentro del set de datos. En este contexto se desea definir el concepto de comando mental, de tal forma que se adapte de manera precisa a la actividad cerebral realizada; la aclaración del concepto de comando mental es relevante e importante, ya que en múltiples publicaciones anteriores se ha utilizado de diversas formas sin exponer una definición clara que le permita reconocerse como una actividad cerebral específica.

Tabla 2. Resultados de validaciones cruzadas (5 y 10 subsegmentos).

	Puntaje (x/1)										
No. subconjutos	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	$\bar{x}$
5	0.94	0.93	0.93	0.94	0.94						0.9406
10	0.94	0.94	0.94	0.94	0.93	0.94	0.94	0.94	0.94	0.95	0.9453

Hoy en día la innovación y crecimiento tecnológico y científico requiere discernir de forma concreta las diferentes actividades mentales aplicadas en los sistemas BCI. A continuación se presenta una propuesta para la definición de comando mental, de igual manera será el utilizado para el presente trabajo.

Definición 1 (Comando mental). Se entiende como comando mental, la actividad cerebral orientada a utilizar exclusivamente la intención del usuario como orden para controlar un sistema BCI. Es importante diferenciar la definición otorgada con otras clasificaciones de actividades mentales como lo es la imaginación motora.

La diferencia radica en que los comandos mentales no requieren la imaginación de un movimiento físico-motor [14] , ya que esto podría dificultar su uso para individuos con discapacidades de este tipo; por otra parte un comando mental se concentra en imaginar con intención la tarea que se desea que el sistema realice.

Debido a la gran entropía que presentan las señales EEG de un mismo individuo dependiendo de la hora o el lugar de la toma de datos, se decidió capturar en un mismo momento todos los datos de un individuo, pudiendo así tener la menor entropía posible en los registros de datos de un solo sujeto de prueba.

Para el proceso de captura de datos, se lleva a cabo un cronometrado de tiempo, donde se le indica al sujeto de prueba que piense en un comando mental durante una X cantidad de tiempo, después le sigue un descanso con una duración Y para luego pensar en otro comando mental durante X tiempo. Además se decidió agregar un conjunto de datos aleatorios donde el sujeto de prueba se encuentre realizando una actividad común que no precise de un esfuerzo significativo.

En la Figura 1 se presenta un diagrama explicativo del proceso de la captura de los datos, donde se observa que cada sesión de grabación constó de dos iteraciones en las que el participante imaginaba por 20 segundos a un objeto mostrado en pantalla deslizándose a la derecha, seguidos por 15 segundos de descanso para después imaginar durante 20 segundos el deslizamiento a la izquierda, por último se graba el último descanso de 15 segundos.

3.4. Extracción de características: Patrón espacial común (CSP)

En busca de amplificar la variabilidad de los datos entre clases y minimizarla en la misma clase, se aplica el método CSP. Este método ha sido utilizado previamente en el área de sistemas BCI como método de extracción de características dentro de la imaginación motora por medio de filtrados espaciales. El objetivo del CSP es encontrar la matriz de proyección W, que funciona como matriz de transformación para proyectar los datos originales desde el espacio de las señales de entrada a un nuevo espacio donde las clases de interés se vuelven linealmente separables.

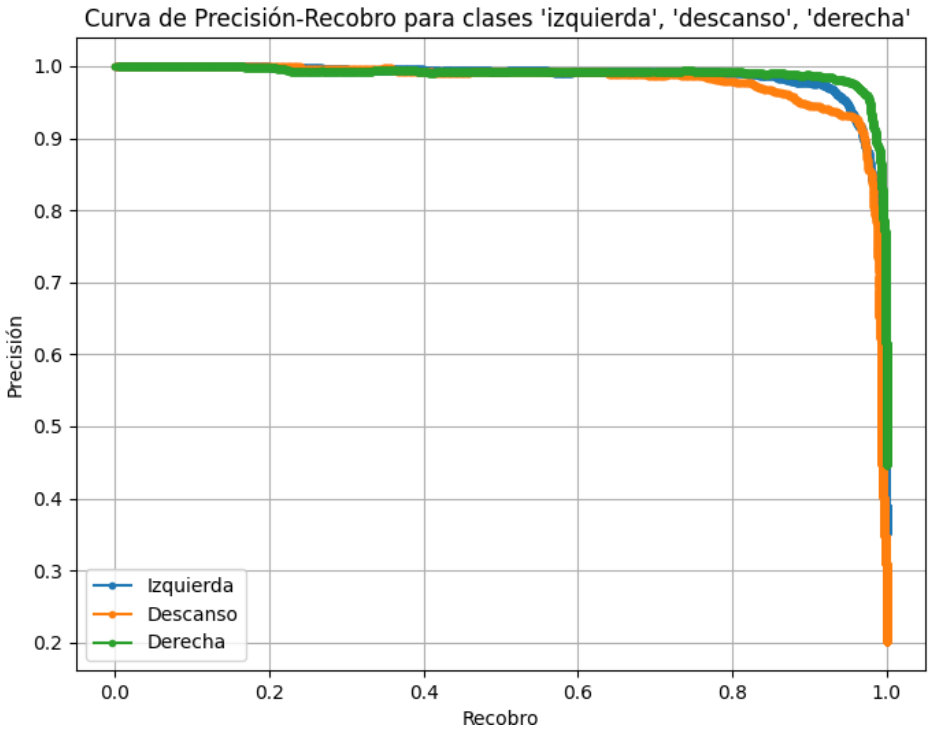


Fig. 3. Curva Precisión-Recobro representativa para la clasificación de comandos mentales con modelo SVM.

Esta transformación se logra encontrando los vectores propios que maximizan la varianza entre clases y minimizan la varianza dentro de cada clase. Este objetivo se puede expresar como se hace en (1) lo que corresponde a un problema de optimización [9]:

$$\max_{W} \frac{(W^T\, S_b\, W)}{(W^T\, S_w\, W)}, \tag{1}$$

donde:

S_b : Representa la matriz de covariancia entre clase.
S_w: Representa la matriz de covarianza interclase.

3.5. Clasificación: Máquinas de soporte vectorial (SVM)

Las SVM son modelos de aprendizaje supervisado que se utilizan ampliamente en problemas de clasificación y regresión. La ventaja principal de las SVM radica en su capacidad para manejar eficientemente conjuntos de datos de alta dimensionalidad y no lineales. Esto las hace especialmente adecuadas para el análisis de señales biológicas, como las señales EEG, que frecuentemente exhiben una alta dimensionalidad y una estructura no lineal [8].

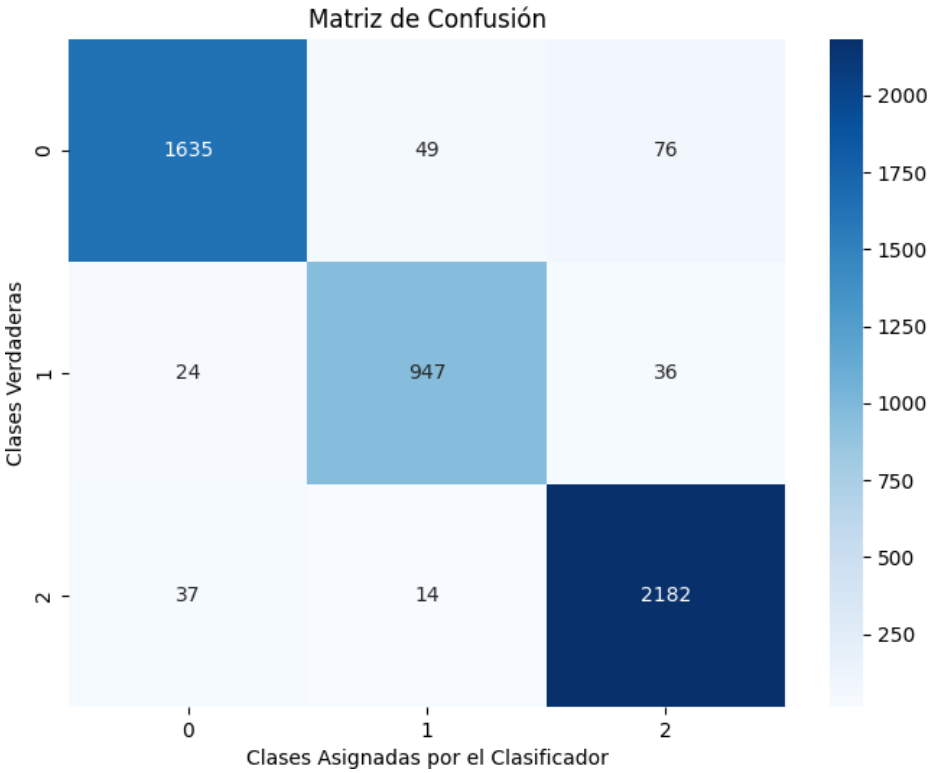


Fig. 4. Matriz de confusión del modelo SVM para la clasificación de comandos mentales. La mayoría de los datos se concentran en la diagonal principal, esto manifiesta que las clases asignadas por el clasificador fueron mayormente correspondientes a la clase verdadera.

El objetivo de las SVM es encontrar el hiperplano con el mayor margen posible que separe las clases de manera lineal, facilitando así la distribución de nuevos datos de entrada en su clase correspondiente, estos datos son representados como un vector n-dimensional [5]. Expresado de forma matemática, las SVM funcionan de la siguiente manera. Dado un conjunto de datos de entrenamiento $(x_1,\ y_1), (x_2,\ y_2), \ldots, (x_n,\ y_n)$ donde x_i es un vector de características y y_i es la etiqueta de clase correspondiente, una SVM lineal busca encontrar el hiperplano de la siguiente forma:

$$w^T x + b = 0, \tag{2}$$

donde w es el vector de pesos y b es el término de sesgo. Dado un nuevo punto de datos x, la SVM predice la clase de la siguiente manera:

$$\text{Clase}(x) = \left\{1 \ \text{ si } \ w^T x + b \geq 0,\ -1 \ \text{ si } \ w^T x + b < 0\ . \right. \tag{3}$$

En el caso de que los datos no sean linealmente separables, se puede utilizar una SVM no lineal que mapea los datos a un espacio de características de mayor dimensión utilizando una función kernel. En este caso, la SVM se vería así:

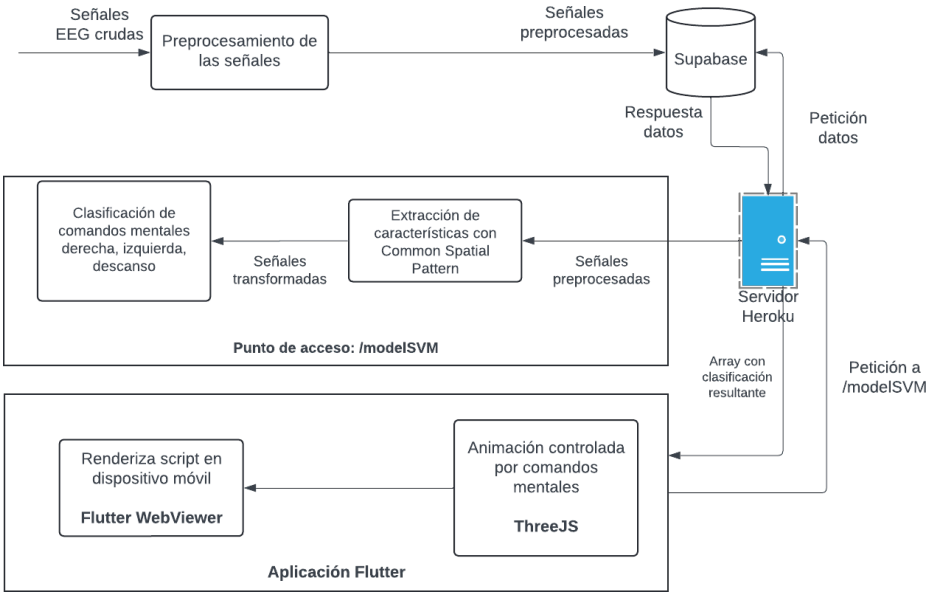


Fig. 5. Arquitectura del sistema desarrollado.

$$\text{Clase}(x) = \begin{cases} 1 \text{ si } \sum_{i=1}^{n} \alpha_i \, y_i \, K(\mathbf{x}_i, \mathbf{x}) + b \geq 0, \end{cases} \tag{4}$$

$$\text{Clase}(x) = \begin{cases} -1 \text{ si } \sum_{i=1}^{n} \alpha_i \, y_i \, K(\mathbf{x}_i, \mathbf{x}) + b < 0, \end{cases} \tag{5}$$

donde $K(x_i, x)K(x_i, x)$ es la función kernel que mide la similitud entre los puntos xi y x, e α_i son los coeficientes de Lagrange obtenidos durante el entrenamiento del modelo. Debido a la gran cantidad de registros analizados dentro de este estudio, así como también al desempeño mostrado por las SVM en trabajos previos, se decidió llevar a cabo la clasificación de comandos mentales por medio de este algoritmo.

Uno de los aspectos fundamentales dentro de este algoritmo es la selección del kernel a utilizar, ya que este determina cómo se mapean los datos de entrada en un espacio dimensional más alto, donde es más probable que sean linealmente separables [12]. Con el objetivo de clasificar de forma precisa los comandos mentales registrados, se optó por el uso de un kernel de función de base radial (RBF por sus siglas en inglés) debido a su notoria aportación al desempeño del algoritmo frente a señales biométricas como las EEG [2].

3.6. Kernel de función de base radial (RBF)

Al seleccionar un kernel RBF se deben consideran un par de parámetros que tienen la capacidad de optimizar el desempeño del modelo, estos son C y γ. Por un lado C se comporta como un parámetro de regularización de la SVM, así pues, controla el balance

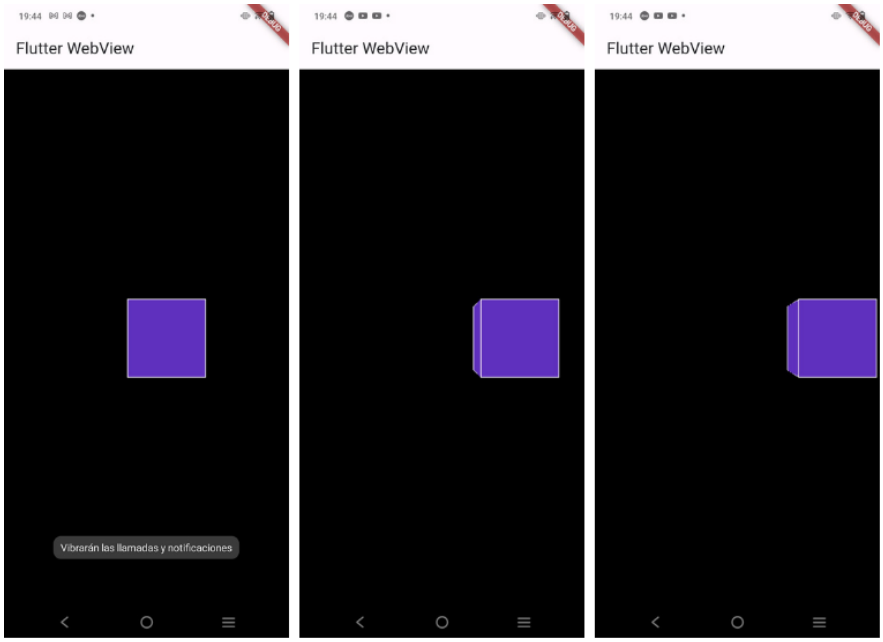


Fig. 6. Capturas de pantalla de la aplicación desarrollada para la visualización del funcionamiento del modelo. En las imágenes, el cubo responde al comando 'derecha'.

entre la maximización del margen y la minimización del error de clasificación en el conjunto de entrenamiento; este mismo penaliza a las clasificaciones incorrectas por lo que un valor alto de C permite que el modelo clasifique más puntos de entrenamiento correctamente incluso si esto significa generar un margen de separación más pequeño entre clases. De manera contraria, un bajo valor para C representa la priorización de un margen más grande aún cuando la precisión de la clasificación de los puntos de entrenamiento se vea comprometida [15].

Definir γ requiere de cuidado para evitar un sobreajuste o un subajuste, ya que este controla la forma en que se propaga la influencia de un punto de datos individual. Especificando, un valor pequeño de γ resulta en fronteras de decisión y modelos más suavizados, lo que se traduce a una transición más gradual entre las regiones clasificadas como una clase u otra y como es de esperarse un valor alto de γ figura el efecto contrario para el modelo [15]. En la Figura 2 se logra visualizar cómo entre menor es C, mayor es el margen que divide a las clases y un menor γ genera regiones de clase más amplias y suavizadas.

4. Resultados

Lograr una clasificación precisa de señales EEG mediante algoritmos de inteligencia artificial sigue siendo un campo de estudio al día de hoy. Este trabajo propone la implementación del método CSP para extracción de características, cuya salida será clasificada por medio de una máquina de soporte vectorial.

En esta sección se hablará de los hiperparámetros seleccionados para el modelo, los resultados de precisión, F1 y recobro obtenidos, así como también se mostrará la arquitectura del sistema desarrollado para la aplicación de esta clasificación en dispositivos móviles.

4.1. Clasificación: Hiperparámetros y resultados SVM

Una vez que la petición al servidor para recuperar los datos de entrada es ejecutada, el conjunto de datos se divide en columnas de atributos y la columna con las etiquetas de clasificación (X = atributos, Y = etiquetas). Seguido de esto, se comienza la extracción de características, donde se calculan las matrices de covarianza para cada clase, calculando entonces la proyección CSP. Esta proyección debe ser aplicada a X, con el fin de obtener los datos transformados que representan los datos de entrada para nuestra SVM (X_csp).

Naturalmente, el conjunto de datos es dividido en datos de entrenamiento y prueba, asignando el 80 % y 20 %. Dado que los datos fueron procesados por el método CSP que busca la maximización de variabilidad entre clases y minimización de variabilidad entre datos de una misma clase, se experimentó con un valor de $C = 2$ relativamente alto para el modelo SVM, mientras que para el hiperparámetro γ se buscó un valor pequeño que ampliara el área de las clases.

La Tabla 1 muestra los hiperparámetros utilizados y los resultados obtenidos por el modelo. Una curva de precisión-recobro permite evaluar el rendimiento de un modelo de clasificación, especialmente cuando las clases están desbalanceadas. Esta curva muestra cómo varía la precisión del modelo en función del recobro (también conocido como sensibilidad o tasa de verdaderos positivos) al cambiar el umbral de decisión del clasificador.

1. **Precisión.** La precisión se refiere a la proporción de instancias clasificadas correctamente como positivas entre todas las instancias clasificadas como positivas por el modelo. Una alta precisión indica que el modelo no clasifica erróneamente muchas instancias como positivas [3].
2. **Recobro.** El recobro indica la proporción de instancias positivas que fueron clasificadas correctamente como positivas por el modelo. Un alto recobro significa que el modelo identifica correctamente la mayoría de las instancias positivas [4].

En la Figura 3. se presenta la curva de precisión-recobro obtenida del modelo construido en esta investigación. Para verificar que no se trata de un modelo sobreajustado, se sometió también a la técnica de validación cruzada, La Tabla 2 muestra los resultados obtenidos. Con el fin de visualizar el desempeño del modelo, se opta por graficar una matriz de confusión, ya que logra proporcionar un resumen claro de la clasificación realizada por el modelo en comparación con las clasificaciones reales. Las filas representan a las clases reales y las columnas a las clases propuestas por el clasificador, se busca que la diagonal principal de la matriz destaque. En la Figura 4 se muestra la matriz de confusión obtenida con los resultados de las predicciones del modelo.

4.2. Arquitectura del sistema

Uno de los principales aportes del presente artículo es la conexión del modelo desarrollado con una aplicación funcional para dispositivos Android, el cual es capaz de controlar un objeto de forma asíncrona con base a las señales previamente clasificadas. La aplicación fue probada en un celular de gama baja con las siguientes características:

- **Memoria RAM**. 3GB.
- **Memoria interna**. 32GB.
- **Procesador**. Octa core.
- **Marca**. VIVO.
- **Modelo**. Y01.
- **Sistema operativo**. Funtouch OS.

La planificación de la arquitectura de los sistemas BCI constituye una fase crucial en su desarrollo. Para adaptar este proyecto a dispositivos móviles, se incorporaron múltiples tecnologías, las cuales se describen a continuación.

- **Supabase**. Para el almacenamiento de los .csv contenidos de las señales EEG capturadas.
- **Heroku**. Utilizado como host para la API.
- **ThreeJS**. Para la animación del objeto a controlar con comandos mentales.
- **Flutter**. Para renderizar la aplicación en un dispositivo móvil.

La Figura 5. muestra la arquitectura diseñada para la comunicación entre la API, modelo e interfaz del sistema desarrollado. En la Figura 6. se presentan capturas de pantalla de la aplicación ejecutándose en un dispositivo Android.

5. Conclusiones y discusión

La aplicación del método CSP para la extracción de características parece potencializar el desempeño de las máquinas de soporte vectorial clasificando señales EEG, dando resultados mayores a 0.9/1 en distintos tipos de evaluación del modelo (precisión, recobro, F1). Trabajos del autor anteriores donde no se aplica CSP reducen a un desempeño promedio de 0.7 en precisión. Se considera que debido a que el método CSP amplía la variabilidad entre clases, se da la oportunidad de probar con hiperparámetros del modelo SVM no convencionales o que comúnmente llevan a sobreajustes o subajustes.

Debido a la creciente importancia y diversificación de los sistemas BCI se invita a la comunidad académica a definir y diferenciar actividades cerebrales que son comúnmente englobadas a otras, pues las aplicaciones de este tipo de tecnología crece día a día, y estos conceptos habilitan nuevos campos de investigación, diseño e implementaciones de sistemas BCI.

Cabe mencionar que si bien se presentó una propuesta para la arquitectura del sistema, el foco de atención se dirigió al procesamiento y clasificación de los datos, por lo que, aunque resultó funcional, se considera que el desempeño de la aplicación se podría ver beneficiada por un rediseño futuro de la arquitectura propuesta.

Referencias

1. Amazon Web Services: ¿Qué es una interfaz de programación de aplicaciones (API)? (2024) aws.amazon.com/es/what-is/api
2. Bousseta, R., Tayeb, S., Ouakouak, I. E., Gharbi, M., Regragui, F., Himmi, M. M.: EEG efficient classification of imagined hand movement using RBF kernel SVM. In: 11th International Conference on Intelligent Systems: Theories and Applications, vol. 113, pp. 1–6 (2016) doi: 10.1109/sita.2016.7772278
3. C3.ai: Precision in machine learning (2024) c3.ai/glossary/machine-learning/precision/
4. C3.ai: Recall in machine learning (2024) c3.ai/glossary/machine-learning/recall/
5. Chatterjee, R., Bandyopadhyay, T.: EEG based motor imagery classification using SVM and MLP. In: 2nd International Conference on Computational Intelligence and Networks, pp. 84–89 (2016) doi: 10.1109/cine.2016.22
6. Dokare, I., Kant, N.: Performance analysis of SVM, kNN and BPNN classifiers for motor imagery. International Journal of Engineering Trends and Technology, vol. 10, no. 1, pp. 19–23 (2014) doi: 10.14445/22315381/ijett-v10p205
7. Eskandar, S.: Introduction to RBF SVM: A powerful machine learning algorithm for non-linear data. Medium (2023) medium.com/@eskandar.sahel/introduction-to-rbf-svm-a-powerful-machine-learning-algorithm-for-non-linear-data-1d1cfb55a1a
8. Ganaie, M. A., Tanveer, M., Jangir, J.: EEG signal classification via pinball universum twin support vector machine. Annals of Operations Research, vol. 328, no. 1, pp. 451–492 (2022) doi: 10.1007/s10479-022-04922-x
9. Gaur, P., Gupta, H., Chowdhury, A., McCreadie, K., Pachori, R. B., Wang, H.: A sliding window common spatial pattern for enhancing motor imagery classification in EEG-BCI. IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1–9 (2021) doi: 10.1109/tim.2021.3051996
10. Kawala-Sterniuk, A., Browarska, N., Al-Bakri, A., Pelc, M., Zygarlicki, J., Sidikova, M., Martinek, R., Gorzelanczyk, E. J.: Summary of over fifty years with brain-computer interfaces—a review. Brain Sciences, vol. 11, no. 1, pp. 43 (2021) doi: 10.3390/brainsci11010043
11. MathWorks: Introducción a support vector machine (SVM). Support Vector Machine (2024) la.mathworks.com/discovery/support-vector-machine.html
12. Mustafa-Abdullah, D., Mohsin-Abdulazeez, A.: Machine learning applications based on SVM classification a review. Qubahan Academic Journal, vol. 1, no. 2, pp. 81–90 (2021) doi: 10.48161/qaj.v1n2a50
13. Peksa, J., Mamchur, D.: State-of-the-art on brain-computer interface technology. Sensors, vol. 23, no. 13, pp. 6001 (2023) doi: 10.3390/s23136001
14. Saibene, A., Caglioni, M., Corchs, S., Gasparini, F.: EEG-based BCIs on motor imagery paradigm using wearable technologies: A systematic review. Sensors, vol. 23, no. 5, pp. 2798 (2023) doi: 10.3390/s23052798
15. Scikit Learn: RBF SVM parameters (2024) scikit-learn.org/dev/auto_examples/svm/plot_rbf_parameters.html.
16. Sha'abani, M. N. A. H., Fuad, N., Jamal, N., Ismail, M. F.: kNN and AVM classification for EEG: A review. In: The 11th Annual Energy Conversion Congress and Exposition, vol. 632, pp. 555–565 (2020) doi: 10.1007/978-981-15-2317-5_47

ISSN 1870-4069

IncluAventuras, un cuentacuentos para niños basado en IA generativa

Keren Mitsue Ramírez-Vergara, Asdrúbal López-Chau,
Rafael Rojas-Hernández, Valentin Trujillo-Mora

Universidad Autónoma del Estado de México,
Centro Universitario UAEM,
Ciudad Laboratorio de Investigación en
Ingeniería y Ciencias Aplicadas,
México

kramirezv003@alumno.uaemex.mx,
{alchau, rrojashe, vtrujillom}@uaemex.mx

Resumen. La discapacidad representa una de las problemáticas más significativas en América Latina, impactando a más de 85 millones de personas en la región. A pesar de los esfuerzos considerables para mitigar las desigualdades entre quienes viven con esta condición, se ha prestado escasa atención al fomento de la inclusión, tolerancia y respeto desde las primeras etapas de la vida. En este artículo, se presenta el diseño completo de IncluAventuras, un cuenta-cuentos digital potenciado con inteligencia artificial generativa (IAG). Además de exponer el diseño, se detallan tres tipos de pruebas realizadas al sistema. Estas incluyen pruebas de funcionamiento, de diversidad en los contenidos generados y un análisis cualitativo de las voces utilizadas en las narraciones. Los resultados obtenidos indican que los sistemas basados en IAG, como es el caso de IncluAventuras, pueden constituir un recurso didáctico invaluable tanto para educadores como para investigadores interesados en la aplicación de la IA en el ámbito educativo.

Palabras clave: IA generativa, ChatGPT, educación inclusiva, cuenta-cuentos.

IncluAventuras, a Storytelling Platform for Children based on Generative AI

Abstract. Disability represents one of the most significant issues in Latin America, affecting over 85 million people in the region. Despite considerable efforts to mitigate inequalities among those living with this condition, little attention has been paid to promoting inclusion, tolerance, and respect from early stages of life. This article presents the complete design of IncluAventuras, a digital storytelling platform powered by generative artificial intelligence (GAI). In addition to outlining the design, three types of tests conducted on the system are detailed. These include functionality tests, diversity tests on generated content, and a qualitative analysis of the voices used in the narratives. The results indicate that GAI-based systems, such as IncluAventuras, can serve as invaluable educational resources for educators and researchers interested in the application of AI in the educational field.

Keywords: AI generative, ChatGPT, inclusive education, storytelling.

1. Introducción

La inteligencia artificial generativa (IAG) presenta un avance significativo en el campo de la inteligencia artificial (IA), especialmente en la producción de diversos tipos de contenidos, como los textos. En el ámbito educativo, su aplicación ha aumentado considerablemente a nivel mundial en la era digital [14], ya que posee el potencial de fortalecer las habilidades de lectura y comprensión de textos en los alumnos de los primeros años escolares. Esta tecnología impulsa la transición hacia una educación más inmersiva, dinámica, participativa e inclusiva, enfatizando el papel crucial de docentes y estudiantes como agentes de cambio en esta transformación [3].

La integración de sistemas como ChatGPT en la educación debería impulsar el desarrollo de las capacidades humanas, al mismo tiempo que contribuya positivamente a la reducción de las desigualdades y fomente los valores fundamentales. Narrar cuentos a niños se destaca como una excelente manera de promover estos valores, pues facilita la comprensión del mundo, estimula la imaginación y promueve la resolución de conflictos [7]. En México, se brinda poca atención a personas con discapacidad durante la educación escolarizada. De acuerdo con [12], del 15 % de la población estudiantil que presenta alguna discapacidad, únicamente el 2.85 % recibe una educación especializada.

Además, en América Latina y el Caribe [1], se observan pocos avances en la mejora en los programas de empleo, educación y servicios de salud para personas con discapacidades. Este artículo se enfoca en la educación inclusiva, en respuesta a la creciente preocupación por la eliminación de la discriminación. Para ello, se describe el diseño e implementación de IncluAventuras, un sistema cuenta-cuentos basado en ChatGPT, para la creación automática de cuentos dirigidos a niños de habla hispana.

El resto del artículo se desarrolla en las siguientes secciones. En la sección 2 se muestra una revisión de la literatura. La Sección 3 presenta el diseño de IncluAventuras, mostrando las tecnologías utilizadas, el diseño del prompt y los detalles más importantes de la implementación. En la sección 4 se presentan los resultados del sistema, finalmente, se presentan conclusiones y las referencias.

2. Revisión de literatura

Los cuentos dirigidos a niños destacan conceptos tales como la compasión, la solidaridad y la empatía [2]. Asimismo, buscan fortalecer a los individuos, fomentando su desarrollo personal, su autonomía y su conciencia cívica [8]. El estudio presentado en [2], evidenció que las historias digitales facilitan la comprensión de valores inclusivos y fomentan la empatía entre los niños. En ese estudio se analizaron las reacciones emocionales de 25 niños al leer un cuento digital.

Los resultados obtenidos fueron positivos, destacando el interés y la sensibilidad hacia el personaje principal, así como la comprensión de la empatía y su aplicación en situaciones cotidianas. No solamente se han usado cuentos digitales para estudiar las reacciones de los niños, en [16], se creó un sistema para digitalizar peluches mediante la cámara Web de la computadora. Se observó que los niños pudieron crear sus juguetes digitalizados en unos minutos. Los sistemas basados en IAG se han empleado para potenciar la creatividad, como se evidencia en algunos estudios [5].

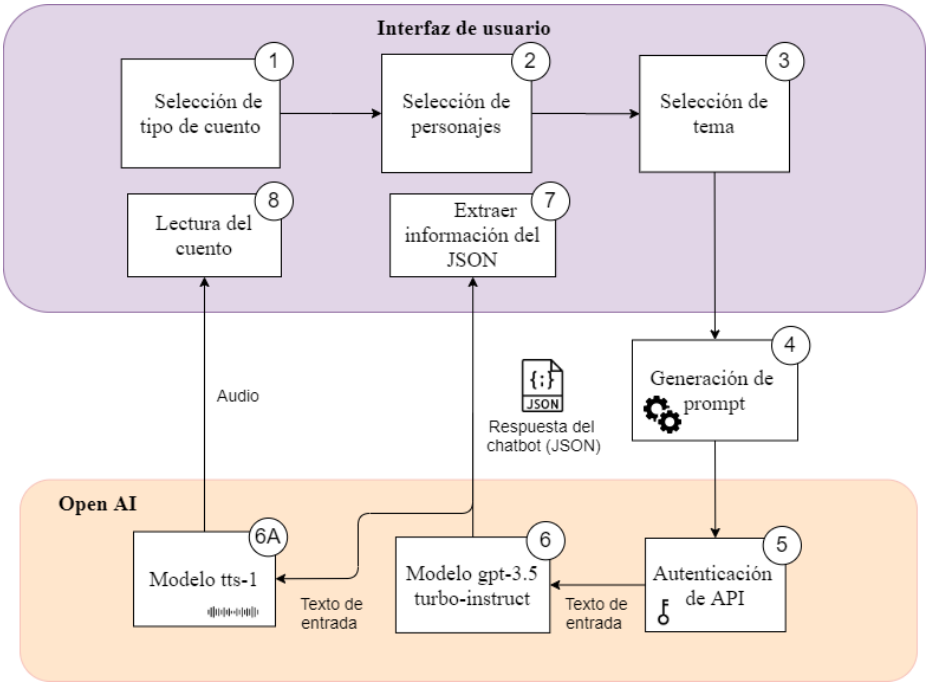


Fig. 1. Arquitectura de IncluAventuras. Elaboración propia.

En [6], se evaluó la creatividad en sistemas de IAG en el ámbito educativo, analizando la flexibilidad, elaboración y originalidad de las respuestas mediante pruebas de aceptación del usuario. Los autores notaron una mejora en el pensamiento divergente y la variedad de perspectivas ofrecidas. La integración cuidadosa de la IAG en la educación creativa promueve una relación simbiótica entre la creatividad humana y la IA. Varios autores consideran que la IA constituye una herramienta valiosa para enriquecer las ideas de escritores humanos, tal como se plantea en [10].

En [17], se remarca que ChatGPT tiene una función educativa clave al facilitar el acceso al conocimiento, generar contenido y fomentar la inclusión educativa, abordando también desafíos éticos como la transparencia y la responsabilidad. En [9], se descubrió que el uso de la aplicación “Kids Story Builder” fortalece la comprensión y la conexión emocional de los niños con ellos mismos y sus familias.

Además, se observó que esta tecnología fomenta el pensamiento narrativo durante la creación de historias. En [4], se sugiere diseñar tecnologías que prioricen la toma de decisiones sobre la imitación humana, ofreciendo oportunidades para que los niños adopten diversas perspectivas y recibiendo retroalimentación significativa con cada acción. Además, respaldan actividades recreativas para reforzar el aprendizaje tecnológico. Pese a la evidencia presentada en los estudios anteriores, también se han identificado diversas desventajas en el uso de la IAG. Por ejemplo, el sesgo en los resultados es una preocupación, y los contenidos generados por la IAG, al basarse en textos o datos recopilados de internet, pueden propiciar el plagio [6].

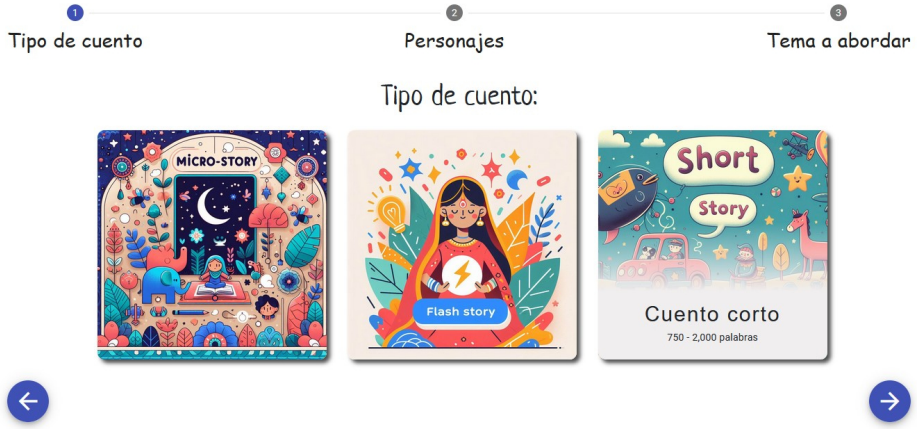


Fig. 2. Interfaz de IncluAventuras para selección del tipo de cuento (captura de pantalla). Elaboración propia.

La revisión de la literatura no reveló implementaciones específicas de la IAG en la educación inclusiva, pero se encontraron guías y recomendaciones para garantizar un impacto positivo. En [13], se ofrecen soluciones para el desafío de la educación inclusiva, asistiendo a profesores, profesionales y responsables políticos en la utilización efectiva de la IA y las nuevas tecnologías para fomentar la inclusión social en la educación. Se resalta el potencial de las tecnologías emergentes para transformar la educación, crear entornos inclusivos y fomentar la colaboración en el aprendizaje [13]. En [15], se proporciona una guía para adoptar la tecnología, enfatizando la necesidad de promover la inclusión social. Se menciona que es fundamental incorporar mecanismos de protección y brindar actualizaciones periódicas a los programadores, fomentar una comprensión ética de la IA entre los estudiantes y guiar a los profesores en la integración de valores éticos en la educación.

3. Materiales y métodos

La arquitectura general del cuenta-cuentos IncluAventuras se resume de manera gráfica en la figura 1. En ella, se puede apreciar tres bloques principales. El primero es la API de Open AI, que ofrece comunicación con ChatGPT y el modelo de tranformación de texto a voz (TTS). El segundo bloque es la interfaz de usuario, con la que el usuario interactua. El tercer componente es un prompt que se diseñó cuidadosamente para producir los mejores resultados.

IncluAventuras fue desarrollado usando TypeScript como lengua je de programación y el framework Angular pare el front-end, es decir, para la creación de interfaces de usuario del sistema. Para proporcionar la funcionalidad de generar historias, se integró ChatGPT en el cuenta-cuentos. Esto debido a las capacidades excepcionales de ChatGPT, que le permiten generar contenido que parece escrito por un humano, debido a lo realista en emociones y personajes.

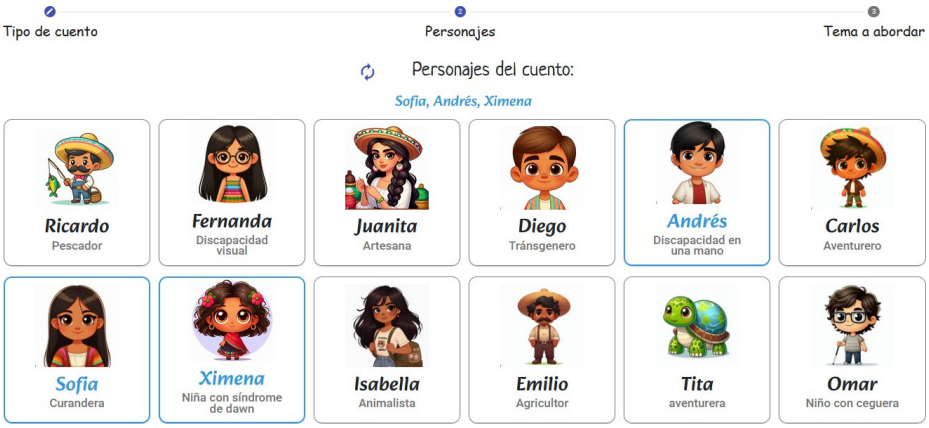


Fig. 3. Interfaz de IncluAventuras para selección de los personajes (captura de pantalla). Elaboración propia.

3.1. Conexión con ChatGPT

Una parte muy importante de IncluAventuras es su integración con la API de OpenAI, la cual ofrece una amplia gama de servicios para el procesamiento del lenguaje natural, la síntesis de voz, la generación de texto y otras funcionalidades. Para acceder a la API de OpenAI, es necesario autenticarse mediante una clave API proporcionada por la plataforma. Es importante tener en cuenta que el uso de esta API está sujeto a tarifas, las cuales dependen del volumen y tipo de solicitudes realizadas. En nuestro caso, utilizamos la versión gratuita para las pruebas realizadas, es decir, la versión GPT-3.5-turbo. Los principales parámetros para configurar ChatGPT son los siguientes.

El número de tokens, que define el límite máximo de tokens que el generador puede producir en una sola solicitud. Un token es una unidad mínima de texto, que generalmente representa una palabra o un símbolo individual; la temperatura, que regula el grado de variabilidad y originalidad en el texto generado. Una temperatura más alta conlleva a respuestas más diversas y creativas, pero también incrementa la probabilidad de obtener respuestas incoherentes o irrelevantes; el mensaje también llamado prompt o promotor de respuestas, que es un texto sin formato en el que se dan instrucciones al modelo. Para IncluAventuras, los valores de los parámetros usados fueron:

a) Número de tokens: 2048,

b) Temperatura: 0.5,

c) Mensaje: Se indicó el rol del sistema, y las instrucciones precisas sobre el contenido.

Como parte del prompt diseñado está el rol que debe de tomar, en nuestro caso es el de un cuenta-cuentos inspirador y creativo para niños. Por motivos de espacio, no se coloca aquí el prompt completo[1].

[1]El prompt usado y código fuente completo de IncluAventuras pueden ser solicitados enviando un e-mail al autor correspondencia de este artículo

Fig. 4. Interfaz de IncluAventuras para selección del tema del cuento (captura de pantalla). Elaboración propia.

El mensaje enviado a la API de OpenAI es esencial para la generación de historias. Se han establecido tres elementos fundamentales en esta estructura que el usuario puede personalizar, asegurando así una generación efectiva de cuentos. Estos elementos son el tipo de cuento, los personajes y el tema. El tipo de cuento puede ser micro-cuento (300 palabras), cuento flash (750 palabras) y cuento corto (2000 palabras). Los personajes fueron creados de forma que algunos de ellos presenten algún tipo de discapacidad. Los temas de los cuentos están centrados en valores éticos e inclusivos, como la tolerancia, cuidado al medio ambiente y respeto.

El niño puede disfrutar de la historia generada en formato de texto, acompañada de imágenes de los personajes que aparecen en el cuento. Además, el cuenta-cuentos posibilita la narración de las historias generadas mediante el uso de voces sintéticas. Esta función se implementó pensando en que los usuarios son niños en edad de aprender a leer. Para lograrlo, se empleó la API TTS (Text To Speech) de OpenAI, que ofrece seis voces integradas que pueden utilizarse para narraciones en varios idiomas, incluyendo el español [11]. A continuación, se presentan los resultados de las pruebas realizadas a IncluAventuras.

4. Resultados

Para evaluar la funcionalidad de IncluAventuras, primero se llevó a cabo una prueba de los cuentos para garantizar su generación adecuada. Posteriormente, se llevó a cabo una evaluación inicial de la diversidad entre los cuentos generados utilizando una medición estadística, específicamente el índice de Jaccard. Además, se examinó el rendimiento de las voces utilizadas en la narración de los cuentos.

4.1. Pruebas de funcionamiento

Se realizaron pruebas a IncluAventuras para garantizar su correcto funcionamiento. Para propósitos demostrativos, se presenta los resultados con los siguientes parámetros de ejecución:

Fig. 5. Ejemplo de cuento generado con IncluAventuras (captura de pantalla). Elaboración propia.

– Se seleccionó de un cuento corto usando la interfaz mostrada en la figura 2.
– Se eligieron como personajes del cuento a Sofía, una curandera; Andrés, un niño con discapacidad motriz en una mano; y Ximena, quien tiene síndrome de Down, la figura 3 muestra la interfaz del cuenta-cuentos.
– El tema seleccionado fue el cuidado del medio ambiente, como se indica en interfaz de la figura 4.

Una vez configurados los parámetros, el cuento se generó en aproximadamente 4950 ms, este se presenta en la figura 5. El audio del cuento se creó utilizando el modelo TTS-1-HD con la voz ”Shimmer”. Para este cuento, el audio tiene una duración de 58 segundos, y el proceso de creación le tomó un total de 10.69 segundos al cuenta-cuentos.

4.2. Diversidad de contenidos generados

Para analizar la diversidad de los contenidos generados con IncluAventuras, se empleó el índice de Jaccard. Este índice, una medida estadística, compara la similitud y diversidad entre dos conjuntos, variando de 0 a 1, donde 0 implica ninguna similitud y 1 significa identidad completa. Previamente a la evaluación de los contenidos de los cuentos, se generaron 30 cuentos de manera aleatoria, de los cuales 8 son microcuentos, 11 son cuentos flash y 11 son cuentos cortos.

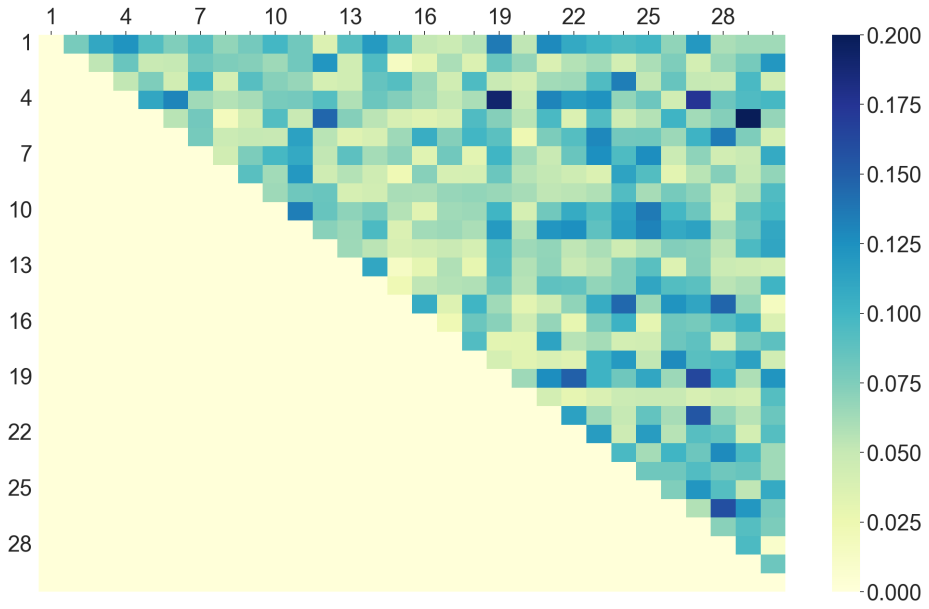


Fig. 6. Índices de Jaccard para cuentos con personajes, tipos y temas aleatorios. Elaboración propia.

Al evaluar estos contenidos con el índice de Jaccard, se observó una amplia variedad en las historias, todas alineadas con las instrucciones del prompt diseñado. La figura 6 presenta gráficamente todos los índices de Jaccard entre los cuentos. El promedio de los índices fue de 0.0359 con una desviación estándar de 0.0427. El valor máximo encontrado fue de 0.2. Por consiguiente, se puede comprobar que los cuentos poseen diversidad entre ellos, es decir, la aplicación genera cuentos únicos y creativos en cada ejecución. Al revisar los cuentos, se destacó que cada uno de ellos transmite un mensaje que fortalece valores y fomenta el respeto hacia los personajes de la historia que presentaban algún tipo de discapacidad o hacia el medio ambiente.

4.3. Análisis cualitativo de la reproducción de cuentos

Basándonos en los 30 cuentos previamente creados, se generaron los audios correspondientes utilizando las 6 voces disponibles del modelo TTS de OpenAI: Alloy, Fable, Echo, Onyx, Nova y Shimmer, generando así 5 audios para cada una de estas voces. Con el objetivo de verificar el desempeño de las voces, se analizaron diversos aspectos que se describen a continuación.

Tiempos de generación. Al analizar los tiempos de respuesta de la API para la generación de los cuentos, se observó que el tiempo de creación tiende a aumentar con la longitud del cuento. Sin embargo, se destacó que en varias ocasiones la respuesta de la API para los cuentos flash fue más rápida en comparación con otros tipos de cuentos, lo que sugiere que el tiempo de respuesta no depende exclusivamente de la longitud del cuento.

Evaluación de tono de voz. Al examinar los tonos de las voces, se identificó que tres de las seis voces (Alloy, Nova y Shimmer) exhiben un tono amigable y agradable durante la narración de los cuentos Además, se observó que la voz de Onyx, a pesar de tener una tonalidad seria, está especialmente adecuada para la narración de cuentos o historias debido a su formalidad que se ajusta a este tipo de narrativa.

Variación de entonación. Una entonación adecuada puede cautivar al oyente, evocar emociones y facilitar la comprensión del contenido. Al analizar este aspecto, se destaca la voz de Onyx por su notable variación de entonación y pausas adecuadas, enriqueciendo la narrativa del cuento. Sin embargo, otras voces presentan pausas breves y una entonación menos eficaz, como en el caso de Nova, lo que puede resultar en una narración más lineal y menos cautivadora, afectando el interés de los niños al escuchar el audio.

Velocidad del habla. En el análisis realizado, se apreció que la voz Nova resulta ineficiente debido a su ritmo variable, lo que dificulta la comprensión del cuento por parte del infante. En cuanto a las voces Fable y Echo mantienen una velocidad constante, a pesar de ser una narración demasiado simple. En contraste, la voz Onyx se destaca por su velocidad de habla, la cual está perfectamente adaptada para la narración de cuentos, proporcionando una experiencia auditiva más envolvente y atractiva.

Calidad de pronunciación. Se reconoció que la voz Nova mostró una pronunciación notablemente deficiente, con episodios de cambios de idioma, lo que indica una falta de estabilidad para mantener el español como idioma principal. Además, se determinó que en todas las voces los nombres de los personajes no fueron pronunciados correctamente. En cambio, la voz con una mejor pronunciación resultó ser Onyx, aunque también enfrentó dificultades al pronunciar los nombres.

5. Conclusiones

En este artículo, se presentó el desarrollo y pruebas realizadas a IncluAventuras, un cuenta-cuentos digital orientado a niños de habla hispana, cuyo objetivo principal es promover desde una edad temprana la inclusión de personas con discapacidad. IncluAventuras produce relatos que cuentan con la participación de personajes infantiles que presentan diversas discapacidades, y siempre incorpora un mensaje positivo. Esto contribuye a brindar una experiencia enriquecedora que fomenta valores en los niños. Se llevaron a cabo pruebas de funcionamiento, análisis de diversidad de contenido y evaluación cualitativa de voz en IncluAventuras.

Se encontró que existe una latencia leve en la generación de los cuentos, pero más acentuada en la generación de voz para las narrativas. Los resultados demostraron una baja similitud entre los cuentos generados, lo cual fue confirmado por el índice de Jaccard. Esto asegura que al utilizar el sistema, los cuentos producidos no se repitan. Además, se examinaron la entonación, velocidad y calidad de pronunciación de las voces sintéticas, identificando áreas de mejora en la tecnología actual para la generación de audio en la narrativa de historias. Basándonos en estos resultados, se concluye que IncluAventuras es una propuesta innovadora y posiblemente beneficiosa para educadores comprometidos con la educación inclusiva.

Como trabajo futuro, se tiene la intención de desarrollar una versión del sistema que pueda operar sin conexión a internet, con el objetivo de beneficiar a poblaciones de escasos recursos. Además, se contempla la integración de un sistema de retroalimentación, permitiendo a IncluAventuras generar cuentos similares a aquellos que hayan sido más apreciados por los niños.

Referencias

1. Banco Mundial: Rompiendo barreras - Inclusión de las personas con discapacidad en América Latina y el Caribe (2021) www.bancomundial.org/es/region/lac/publication/rompiendo-barreras
2. Bratitsis, T., Ziannas, P.: From early childhood to special education: Interactive digital storytelling as a coaching approach for fostering social empathy. Procedia Computer Science, vol. 67, pp. 231–240 (2015) doi: 10.1016/j.procs.2015.09.267
3. Droubi, S., Galamba, A., Fernandes, F. L., de-Mendonça, A. A., Heffron, R. J.: Transforming education for the just transition. Energy Research and Social Science, vol. 100, pp. 103090 (2023) doi: 10.1016/j.erss.2023.103090
4. Druga, S., Vu, S. T., Likhith, E., Qiu, T.: Inclusive AI literacy for kids around the world. In: Proceedings of FabLearn, pp. 104–111 (2019) doi: 10.1145/3311890.3311904
5. Haase, J., Hanel, P. H.: Artificial muses: generative artificial intelligence chatbots have risen to human-level creativity. Journal of Creativity, vol. 33, no. 3, pp. 100066 (2023) doi: 10.1016/j.yjoc.2023.100066
6. Habib, S., Vogel, T., Anli, X., Thorne, E.: How does generative artificial intelligence impact student creativity? Journal of Creativity, vol. 34, no. 1, pp. 100072 (2024) doi: 10.1016/j.yjoc.2023.100072
7. Iruri-Quispillo, S., Villafuerte-Alvarez, C. A.: Importancia de la narración de cuentos en la educación. Comuni@cción: Revista de Investigación en Comunicación y Desarrollo, vol. 13, no. 3, pp. 233–244 (2022) doi: 10.33595/2226-1478.13.3.720
8. Juppi, P.: Engagement and empowerment. Digital storytelling as a participatory media practice. Nordicom Review, vol. 39, no. 2 (2017)
9. Kalantari, S., Rubegni, E., Benton, L., Vasalou, A.: "When I'm writing a story, I am really good" Exploring the use of digital storytelling technology at home. International Journal of Child-Computer Interaction, vol. 38, pp. 100613 (2023) doi: 10.1016/j.ijcci.2023.100613
10. Li, R.: A "Dance of storytelling": Dissonances between substance and style in collaborative storytelling with AI. Computers and Composition, vol. 71, pp. 102825 (2024) doi: 10.1016/j.compcom.2024.102825
11. OpenAI: Text to speech (2023) platform.openai.com/docs/guides/text-to-speech
12. Pozas, M., Trujillo, C. J. G., Letzel-Alt, V.: Mexican school students' perceptions of inclusion: A brief report on students' social inclusion, emotional well-being, and academic self-concept at school. Frontiers in Education, vol. 8 (2023) doi: 10.3389/feduc.2023.1069193
13. Salas-Pilco, S. Z., Xiao, K., Oshima, J.: Artificial intelligence and new technologies in inclusive education for minority students: A systematic review. Sustainability, vol. 14, no. 20, pp. 13572 (2022) doi: 10.3390/su142013572
14. Sanabria-Navarro, J. R., Silveira-Pérez, Y., Pérez-Bravo, D. D., de-Jesús-Cortina-Núñez, M.: Incidences of artificial intelligence in contemporary education. Comunicar, vol. 77, pp. 97–107 (2023)
15. Sijing, L., Lan, W.: Artificial intelligence education ethical problems and solutions. In: 13th International Conference on Computer Science and Education, pp. 1–5 (2018) doi: 10.1109/iccse.2018.8468773

16. Tseng, T., Murai, Y., Freed, N., Gelosi, D., Ta, T. D., Kawahara, Y.: Plushpal: storytelling with interactive plush toys and machine learning. In: Proceedings of the 20th Annual ACM Interaction Design and Children Conference, pp. 236–245 (2021) doi: 10.1145/3459990.3460694
17. Yu, H.: The application and challenges of ChatGPT in educational transformation: new demands for teachers' roles. Heliyon, vol. 10, no. 2, pp. e24289 (2024) doi: 10.1016/j.heliyon.2024.e24289

ISSN 1870-4069

Comparación de modelos para la clasificación automática de temáticas en tuits de comunicación pública de la ciencia en español de México

Alec Sánchez-Montero, Gemma Bel-Enguix,
Sergio Luis Ojeda-Trueba

Universidad Nacional Autónoma de México,
México

alecm@comunidad.unam.mx,
{gbele, sojedat}@iingen.unam.mx

Resumen. En el contexto mexicano, los estudios exhaustivos sobre la comunicación pública de la ciencia (CPC) a través de redes sociales son una tarea pendiente hasta la fecha. Como respuesta a este vacío, se propone este trabajo desde la perspectiva del procesamiento del lenguaje natural (PLN). En concreto, el estudio apunta al desarrollo y la evaluación de la clasificación automática de tuits de CPC publicados en México mediante el entrenamiento de distintos modelos de aprendizaje automático, incluidos algoritmos clásicos y modelos basados en transformers. Con base en un corpus etiquetado manualmente, se evalúan y se comparan varios enfoques para identificar y clasificar automáticamente las áreas temáticas en los tuits de CPC. Los resultados muestran que los modelos clásicos como support vector machine mantienen un rendimiento sólido, mientras que los transformers ofrecen alternativas prometedoras para esta tarea.

Palabras clave: Procesamiento del lenguaje natural, clasificación multietiqueta de texto, comunicación pública de la ciencia.

Model Comparison for Automatic Topic Classification in Mexican Spanish Tweets on Public Communication of Science

Abstract. In the Mexican context, comprehensive studies on public communication of science (PCS) through social networks are a pending task to date. In response to this gap, this work is presented from the perspective of natural language processing (NLP). Specifically, the study aims at developing and evaluating the automatic classification of PCS tweets published in Mexico by training different machine learning models, including classical algorithms and models based on transformers. On the basis of a manually labeled corpus, several approaches to automatic identification and classification of thematic areas in PCS tweets are evaluated and compared. The results show that classical models such as support vector machine maintain a robust performance, while transformers offer promising alternatives for this task.

Keywords: Natural language processing, multi-label classification of text, public communication of science.

1. Introducción

En el contexto de la denominada "Cuarta Revolución Industrial", caracterizada por el amplio uso de las tecnologías digitales, redes y plataformas sociales se han reafirmado como espacios virtuales propicio para la comunicación de conocimientos científicos de distintas disciplinas. En estas redes y plataformas, la información científica se destina tanto a expertos como a audiencias no especializadas. En particular, Twitter —ahora 𝕏— se ha convertido en un popular espacio de comunicación digital masiva, donde se comparten conocimientos científicos, se discuten descubrimientos y se promueve el diálogo sobre temas científicos de distintas disciplinas. Este tipo de interacciones entre investigadores, divulgadores, entusiastas de la ciencia y el público general proporciona una conveniente fuente de datos para explorar y comprender los fenómenos relativos a la comunicación de la ciencia, en los cuales el lenguaje natural tiene una función central.

Específicamente, la comunicación pública de la ciencia (CPC) implica un proceso de difundir y divulgar conocimientos científicos hacia el público en general en un canal bidireccional, fuera del ámbito especializado entre pares o expertos. En este sentido, los comunicadores de la ciencia se dirigen a personas que no poseen formación especializada en ciencias para construir un diálogo en torno a los descubrimientos, los avances y los debates científicos. En años recientes, esta actividad se ha expandido más allá de los canales tradicionales, como las revistas de divulgación, hacia los entornos digitales de interacción social. Para el caso de la CPC en Twitter/𝕏, los textos publicados, conocidos como "tuits" o posts, son breves y fragmentados, influenciados por factores como la interactividad de la plataforma y las limitaciones de espacio [1, 2].

El uso de Twitter en México es bastante significativo, puesto que la base de usuarios excede los 17 millones. Esto sitúa a México entre los primeros 10 países con mayor cantidad de usuarios activos a nivel mundial y, además, como el segundo lugar en Latinoamérica, sólo por detrás de Brasil; asimismo, se trata del país hispanohablante con mayor presencia global[18]. Esta plataforma es especialmente útil para la investigación en lingüística y en procesamiento del lenguaje natural (PLN), pues se pueden compilar corpus lingüísticos a partir de la amplia cantidad de datos generados diariamente y, de ese modo, se pueden estudiar distintos aspectos lingüísticos en función de los objetivos que se persigan en la investigación [20].

Desde una perspectiva de PLN y de ciencia de datos, el estudio de la CPC en México a través de Twitter permite analizar distintas facetas del fenómeno, por ejemplo: la interacción entre los usuarios y los comunicadores científicos, la cobertura de la información en función de los temas, las disciplinas o áreas en las que pueden clasificarse los textos, la influencia de comunicadores particulares, la percepción pública de la actividad científica a nivel nacional e internacional, el impacto del conocimiento científico comunicado en los individuos y en la sociedad, entre otras. Sin embargo, pese al elevado número de usuarios en México, en la revisión de la literatura no se han localizado conjuntos organizados de datos, de libre acceso, en formato de tuits del género CPC publicados en México. Este vacío implica una falta de estudios exhaustivos sobre la CPC en México y, en particular, la ausencia de modelos de aprendizaje automático, o Machine Learning, especializados en la clasificación de tuits según su contenido temático.

En este contexto, el objetivo de este trabajo es desarrollar y evaluar un modelo de clasificación automática basado en un corpus, conformado por tuits de CPC publicados en México, anotado manualmente mediante un sistema multietiqueta conforme a las áreas temáticas abordadas en cada texto del corpus. La metodología propuesta se basa en un enfoque de aprendizaje automático supervisado, en el cual se entrena un modelo computacional a partir de los ejemplos etiquetados previamente por anotadores humanos. En este caso, se lleva a cabo un análisis contrastivo entre algoritmos clásicos de aprendizaje automático, como Support Vector Machine (SVM) y Random Forest Classifier (RFC), y modelos preentrenados, basados en arquitecturas de aprendizaje profundo tipo transformers, como BERT y RoBERTa, con la finalidad de identificar las características más adecuadas en el modelo para una tarea de clasificación automática multietiqueta.

2. La CPC en Twitter/𝕏

En el ámbito de la comunicación científica, se ha generado ambigüedad terminológica entre conceptos como "divulgación científica", "comunicación pública de la ciencia", "alfabetización científica", "periodismo científico", "participación pública en la ciencia", entre otros. A rasgos muy generales, uno de los propósitos de la CPC es acercar el conocimiento científico a un público amplio y no especializado. Al referirse a una CPC en lugar de una divulgación científica, se acentúa la característica bidireccional en el proceso comunicativo.

Según [16], las actividades de CPC como un campo "multi, inter y transdisciplinario que conjunta saberes provenientes de diversas áreas tales como las ciencias naturales, exactas, de la salud, tecnologías, ingenierías y recientemente sociales y humanísticas, así como el manejo de los distintos medios de comunicación y el conocimiento de los diferentes públicos". Desde este enfoque, las actividades de divulgación científica estarían englobadas en las de CPC, como concepto más general.

Un modelo de comunicación científica, como el de la participación del público, encuentra en Twitter/𝕏 un lugar apropiado para su implementación. En este modelo específico se busca generar un diálogo y un compromiso con el público no científico [12]. Por su parte, trabajos previos han estudiado en esta red social fenómenos como la influencia de las interacciones entre usuarios en el interés y la comprensión del público hacia la ciencia [8], el impacto de determinadas figuras clave en la comunicación científica [7], la relación entre determinados temas científicos y el interés público hacia la ciencia mediante métricas de interacción [10], la diseminación del vocabulario científico [19] o el papel de las comunidades educativas para promocionar la comunicación científica [9].

En años recientes, se ha notado cómo el uso de plataformas digitales como Twitter/𝕏 para la comunicación científica despeja las líneas divisorias entre la comunidad "especializada" y el público general [14]. La investigación sobre la comunicación científica en Twitter/𝕏 es un tema incipiente, aunque cada vez se destaca más su relevancia como escenario para la circulación de la información científica y para la interacción entre científicos y audiencias no científicas.

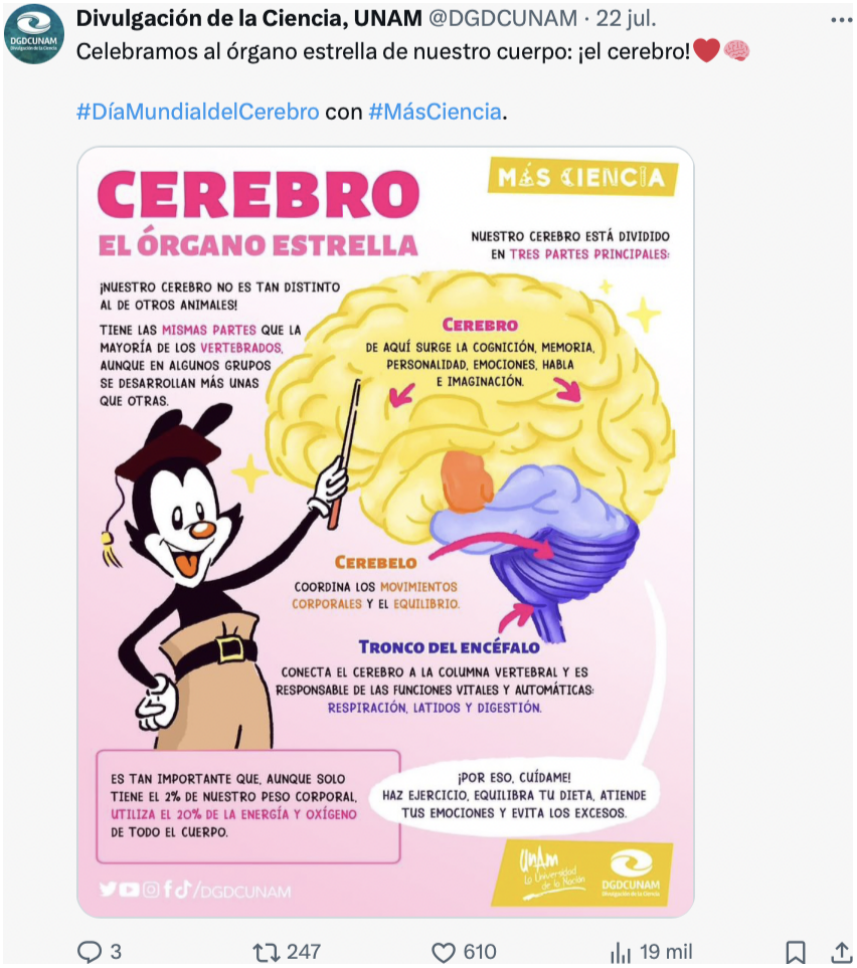


Fig. 1. Ejemplo de tuit de CPC en español mexicano (Fuente: Twitter/𝕏).

Asimismo, la comunidad especializada ha comenzado a centrarse más en esta red social, pues la han reconocido como una oportunidad para investigar, expandir y debatir el conocimiento científico [4, 6, 13]. Según el modelo de participación del público [12], Twitter/𝕏 actúa como una plataforma global para democratizar el conocimiento, aunque, al mismo tiempo, puede contribuir a multiplicar la desinformación y el conocimiento falaz, como se vio a lo largo de la pandemia por COVID-19 [5]. Con respecto al contexto mexicano, se ha observado, en una exploración inicial, una amplia diversidad de los temas comunicados en la plataforma.

Las temáticas destacadas incluyen áreas como la biología, la astronomía y las ciencias de la salud, aunque es común encontrar tuits que integran múltiples áreas científicas o temáticas en un mismo texto. Por otra parte, se debe señalar que los tuits de CPC suelen servir de complemento a otras actividades o enlazar con recursos destinados al público no especializado, ya sean en modalidad física o virtual.

Se incluye la Figura 1 para ilustrar algunas de los principales rasgos de estos tuits, a través de un ejemplo prototípico. Por lo que respecta a los agentes involucrados en la CPC, el conjunto de comunicadores de la ciencia es de carácter heterogéneo. Entre los participantes se puede encontrar tanto instituciones como individuos que comunican los conocimientos científicos. De dichos comunicadores destacan las instituciones de educación superior, así como organizaciones y centros de investigación, publicaciones de divulgación científica, investigadores, estudiantes y divulgadores individuales.

3. Metodología y características del dataset

La metodología de este trabajo se ha desarrollado con base en un pipeline para la clasificación de textos [11]. Este pipeline consiste en las siguientes etapas: 1) recopilación y selección de datos, 2) anotación de los datos, 3) preprocesamiento del texto, 4) extracción de características, 5) selección de las técnicas de clasificación y 6) evaluación. En la primera etapa, se recurrió a la Twitter API v2 para extraer los timelines de una lista de usuarios delimitados mediante la biblioteca Tweepy en un entorno de Python.

Esta lista de usuarios fue el resultado de una investigación para delimitar aquellos perfiles de Twitter relacionados con la CPC en el contexto mexicano. Estos perfiles corresponden a 19 autodenominados "divulgadores", "comunicadores" o "periodistas" científicos, de cuentas individuales e institucionales sobre áreas científicas generales. En otras palabras, las cuentas seleccionadas para este estudio representan una variedad de temáticas científicas generales y no especializadas, que que se corresponde con las áreas del conocimiento divulgadas en el contexto mexicano.

Cabe destacar que estos datos fueron recopilados sin preferencias específicas por un ámbito científico concreto, dado que se ha partido de un escenario con muy poca información cuantitativa y cualitativa respecto al contexto del objeto de estudio. Esta situación trajo consigo una amplia gama de temas en el corpus, desde la astronomía y la física general hasta la genética y la historia de la ciencia, entre otras áreas. En términos de clases del corpus, este tipo de distribución temática al azar puede implicar un desbalance en las clases del corpus.

Sin embargo, se justifica adoptar esta estrategia para reflejar de manera más precisa la distribución natural de etiquetas y clases en el contexto de los tuits de CPC en México, sin manipular los datos para lograr el balance entre clases. Al construir un dataset que refleje la verdadera distribución de clases, se puede identificar áreas específicas donde los modelos pueden tener un rendimiento óptimo y subóptimo, una característica fundamental basada en datos para la investigación de esta emergente área de estudio.

Para constituir un dataset consistente y apropiado para la tarea de clasificación, se delimitó como criterio de homogeneidad fundamental que todos los tuits hubieran sido publicados en español dentro de México, entre enero de 2020 y mayo de 2023, momento en el que se recopilaron de los datos. Tras reunir y estructurar esta información con la Twitter API v2, ocurrieron modificaciones significativas en la plataforma, derivadas del cambio de propietario. La transición del nombre "Twitter" a "𝕏" fue una de ellas. Otro cambio importante, relacionado con el acceso a y la recolección de los datos generados en la plataforma fue la eliminación del acceso gratuito a la API académica,

junto con la introducción de un modelo de pago mensual que comienza en los 100 dólares estadounidenses para acceder a una API con funcionalidades más limitadas, en comparación con la ahora inexistente versión académica. Es relevante considerar esta modificación, puesto que el desarrollo sucesivo de investigaciones relacionadas con esta plataforma podría verse condicionado por limitantes económicas.

Después de eliminar mensajes duplicados, textos de índole personal o de opinión no científica por parte de los autores, el dataset se conformó por 3733 tuits. Este dataset fue tomado como corpus de la investigación para ser etiquetado conforme al contenido temático de cada texto. De acuerdo con la función de tokenización de la biblioteca SpaCy, la cantidad total de tokens en el corpus es de 144,375 y la cantidad de tokens únicos es de 21,830.

Para llevar a cabo la anotación del corpus se seleccionó Argilla, una plataforma de código abierto especializada en el desarrollo de LLM a partir de conjuntos de datos que los usuarios pueden cargar mediante un código de programación a través de un entorno de Python. En esta tarea se buscó identificar y clasificar las áreas temáticas presentes en los tuits del corpus, más allá de procurar que estuvieran balanceadas, con base en una lista de etiquetas predefinidas.

Para anotar cada tuit, se empleó un enfoque de clasificación de texto multietiqueta, es decir, cada tuit podía ser etiquetado con una o más categorías temáticas, contenidas en la lista de etiquetas, según su contenido. Para ello, se optó por la función de "Feedback Dataset" en Argilla, la cual permite la clasificación multietiqueta de registros individuales en un dataset de manera flexible y sencilla.

Como resultado, se obtuvo un conjunto de 18 etiquetas que trató de ser lo más exhaustivo y representativo posible: "astronomía y espacio", "matemáticas", "física", "biología", "medicina y salud", "tecnología", "química", "historia de la ciencia", "ingeniería", "computación", "ciencias de la tierra", "materia y energía", "psicología", "invitación a evento o a recursos", "efeméride", "mujeres en la ciencia", "cultura pop" y "otro". La visualización de estas 18 etiquetas en la interfaz de Argilla[1] se muestra en la Figura 2.

Como instrucción para el etiquetado se indicó leer con atención cada uno de los tuits por separado y, con base en la información del texto y los rasgos contextuales de cada registro, seleccionar todas las etiquetas que describieran el contenido de cada tuit. Una vez terminada la tarea de anotación del corpus, se obtuvieron los resultados presentados en el gráfico de la Figura 3. Como puede observarse en el gráfico, cada tuit puede estar asociado con una o varias etiquetas, por la naturaleza multietiqueta del etiquetado, de forma que la suma de las frecuencias no es igual al total de registros del corpus.

Los resultados del etiquetado revelan la distribución de áreas temáticas predominantes en el corpus, la cual no corresponde a un reparto equilibrado de clases, pues algunas etiquetas mantienen una amplia representación en contraste con otras etiquetas con pocos casos identificados. Entre las etiquetas más frecuentes se encuentran "biología" con 1701 instancias, "invitación a evento o a recursos" con 1664, "física" con 1481, "astronomía y espacio" con 1223 y "medicina y salud" con 755. En contraste, las áreas menos representadas en el corpus son "computación" con 98 instancias, "cultura pop" con 78 y "otro" con 26.

[1] argilla.io

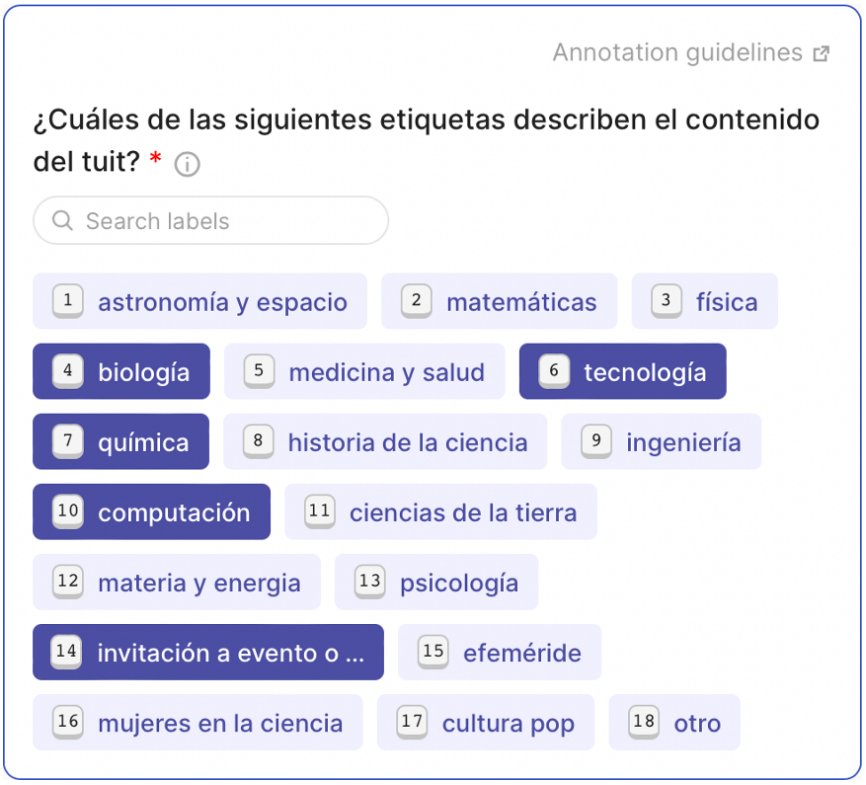


Fig. 2. Visualización de las etiquetas utilizadas para la anotación multietiqueta de las áreas temáticas del corpus en Argilla.

Esta distribución indica la prevalencia de áreas generales (como la biología y la física) y específicas (como la astronomía y temas del espacio) que podrían corresponder a los temas más relevantes en el contexto mexicano de la CPC, al mismo tiempo que las áreas menos representadas podrían ser las menos divulgadas (como el caso de la computación), complemento a otra área temática más general (como el caso de la cultura pop) o no relevante para el estudio de la CPC en Twitter (como la categoría de "otro"). Con este corpus como base para el entrenamiento para el modelo de clasificación automática de tuits, se procedió a las siguientes fases de la metodología.

Se seleccionaron algoritmos clásicos de aprendizaje automático debido a que algunos trabajos previos [15] han demostrado su utilidad para la clasificación de texto con datasets pequeños. A su vez, se buscó contrastar el rendimiento de estos algoritmos en relación con modelos basados en transformers, que representan un estado más avanzado en lo referente a tareas de PLN. Al utilizar los algoritmos clásicos, se consideró necesario preprocesar los textos del corpus. En esta etapa, se recurrió a diversas técnicas de limpieza y normalización de texto, como la eliminación de signos de puntuación, de emoji, de hashtags, de stop words, de hipervínculos, de direcciones de correo electrónico y otros elementos no léxicos, la lematización de palabras y la conversión a minúsculas.

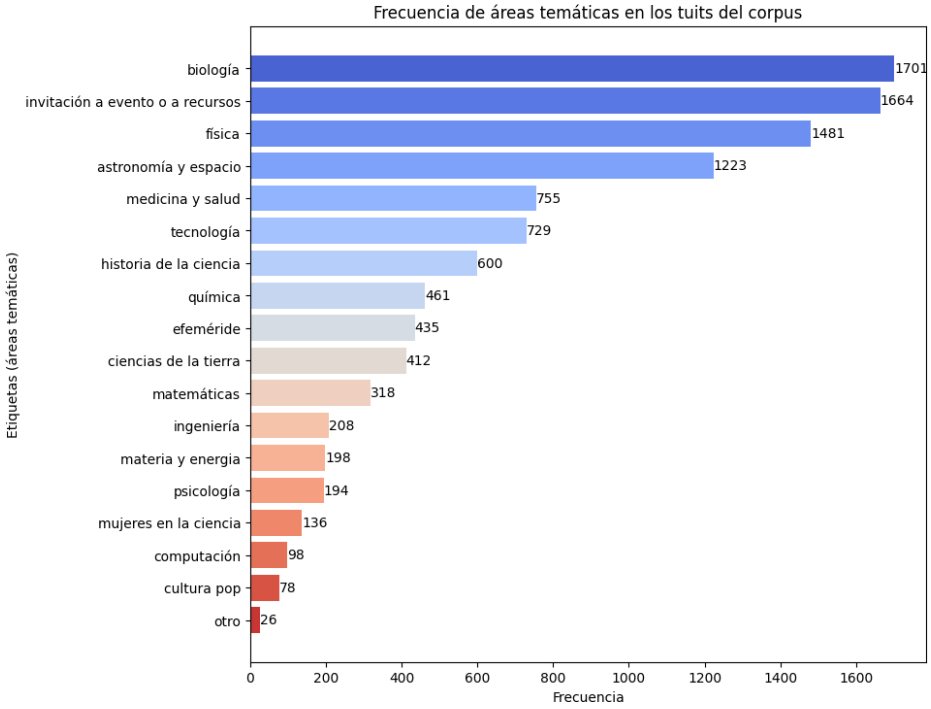


Fig. 3. Distribución de las áreas temáticas en los tuits del corpus.

A continuación, se extrajeron las características de los tuits preprocesados mediante la técnica TF-IDF para obtener representaciones numéricas de las características semánticas y contextuales de los textos, para poder entrenar los algoritmos clásicos de aprendizaje automático seleccionados. Por las características de los modelos basados en arquitecturas transformers, el proceso de entrenamiento y evaluación de los datos se simplificó en comparación con los modelos tradicionales de aprendizaje automático. En otras palabras, como las arquitecturas transformers están diseñadas para trabajar con secuencias de texto de manera eficiente, llevar a cabo una etapa de preprocesamiento exhaustiva se vuelve algo opcional.

En este caso, no se realizó preprocesamiento de los textos y, para la extracción de características, se empleó la biblioteca Transformers de Hugging Face a partir de la representación de los datos en forma de tokens de texto sin procesar, los cuales se proporcionaron como entrada a los modelos para que aprendieran automáticamente las representaciones de características durante el proceso de entrenamiento, como es práctica estándar con este tipo de modelos en PLN.

Por otra parte, ya que se trabajó con un conjunto de datos anotados con 18 etiquetas diferentes, se utilizó un enfoque de codificación de etiquetas conocido como multilabelbinarizer, mediante el uso de la biblioteca Scikit-learn. Esta técnica permite representar adecuadamente las etiquetas multietiqueta en forma de vectores binarios conformados por unos y ceros, donde cada elemento del vector indica la presencia (uno) o ausencia (cero) de una etiqueta particular.

Tabla 1. Resultados de los modelos SVM y RFC en el conjunto de datos de evaluación con base en métricas de precisión, exhaustividad y F1.

Modelo	Precisión	Exhaustividad	F1
SVM	0.89	**0.6**	**0.72**
RFC	**0.9**	0.56	0.69

Este enfoque fue utilizado tanto en los algoritmos clásicos como en los modelos transformers para homogeneizar el entrenamiento del modelo en la clasificación de tuits con múltiples etiquetas. Para la selección de algoritmos clásicos, se optó por dos algoritmos ampliamente utilizados en tareas de clasificación: Support Vector Machine (SVM) y Random Forest Classifier (RFC). En cuanto a los modelos preentrenados basados en arquitecturas transformers, se seleccionaron los siguientes: `distilbert-base-multilingual-cased` [17], `distilroberta-base` [17] y `twitter-xlm-roberta-base-sentiment` [3].

Por lo que respecta a los primeros dos, se trata de versiones ligeras y eficientes de BERT (en el caso de distilbert) y de RoBERTa (en el caso de distilroberta), los cuales han sido preentrenados en en varios idiomas, incluido el español. Finalmente, el modelo `twitter-xlm-roberta-base-sentiment` está específicamente diseñado para tareas de análisis de sentimientos en texto de redes sociales como Twitter y también fue preentrenado en español. Para evaluar estos modelos se emplearon métricas estándar de evaluación de clasificación: precisión, exhaustividad (recall) y F1-score.

4. Entrenamiento y evaluación de los modelos de clasificación automática

Como se mencionó en la sección anterior, para el entrenamiento con los algoritmos clásicos de aprendizaje automático (i.e. SVM y RFC), se llevó a cabo un preprocesamiento de los textos y una extracción de características mediante la técnica de TF-IDF. Asimismo, se empleó la técnica de validación cruzada con 5 folds o iteraciones para evaluar el rendimiento de los modelos con mayor precisión. Esta técnica permite dividir el conjunto de datos para obtener cinco medidas de rendimiento, que después se promedian para obtener una estimación general del desempeño del modelo. La Tabla 1 presenta los resultados detallados de los algoritmos SVM y RFC en el conjunto de datos de evaluación, o test dataset, correspondiente al 20 % del dataset con un random state de 42 para reproducibilidad.

Como puede observarse en estos datos, el modelo RFC tiene una precisión ligeramente mayor (0.9) en comparación con el modelo SVM (0.89). Sin embargo, el modelo SVM tiene una exhaustividad (o recall) más alta (0.6) en comparación con el RFC (0.5). En cuanto al F1-score, medida que combina precisión y exhaustividad, el modelo SVM obtuvo un valor de 0.72, mientras que el RFC obtuvo 0.69. Estos resultados indican un rendimiento muy similar por parte de los algoritmos clásicos de aprendizaje automático seleccionados. Ambos modelos muestran valores de precisión altos, lo cual significa que son capaces de realizar predicciones correctas con una tasa alta.

Tabla 2. Resultados de los modelos transformers con diferentes números de epochs en el conjunto de datos de evaluación.

Modelo	# de epochs	Precisión	Exhaustividad	F1
`distilbert-base- multilingual-cased`	5	**0.83**	**0.60**	**0.64**
	10	**0.83**	0.62	**0.67**
`distilroberta-base`	5	0.74	0.48	0.52
	10	0.73	0.56	0.60
`twitter-xlm- roberta-base-sentiment`	5	0.82	0.59	0.63
	10	0.81	**0.63**	0.65

Por lo que respecta a la exhaustividad y a la puntuación F1, se registran valores por encima del azar. Cabe señalar que estos resultados se refieren al rendimiento general de los modelos y no a las métricas por etiquetas específicas. En cuanto a los modelos basados en arquitecturas transformers, como ya se mencionó, tres modelos preentrenados en textos multilingües fueron seleccionados para la tarea de clasificación automática de texto. Dos de estos modelos corresponden a diferentes versiones del modelo RoBERTa y uno de ellos a una versión más compacta del modelo BERT.

Tanto `distilbert-base-multilingual-cased` como `distilroberta-base` fueron configurados para la tarea de clasificación de secuencias con la biblioteca Transformers, con el problema de una clasificación multietiqueta. Por su parte, el modelo `twitter-xlm-roberta-base-sentiment` fue finamente ajustado para adecuarse a la tarea de clasificación con 18 etiquetas, ya que su arquitectura base comprende la predicción de tres etiquetas, conforme al análisis de sentimientos: positivo, negativo y neutro.

Al igual que se hizo con los modelos de SVM y RFC, para el entrenamiento de los modelos basados en transformers, se dividió el conjunto de datos en una proporción 80-20 (entrenamiento-prueba) con un random state de 42. Además, para las predicciones de los modelos basados en arquitecturas transformers, se definió un umbral de 0.5 para la asignación de etiquetas. Esto significa que una etiqueta se asigna a una instancia de texto si la probabilidad predicha para esa etiqueta es igual o mayor a 0.5. Los resultados de los modelos basados en transformers, en relación con el conjunto de datos de prueba, se muestran en la Tabla 2.

En dicha tabla, se presenta la precisión, la exhaustividad (recall) y la puntuación F1 para cada modelo en torno a un entrenamiento basado en 5 y en 10 epochs. Al comparar estos resultados con los obtenidos por los modelos SVM y RFC, se observa un rendimiento generalmente inferior en términos de precisión, exhaustividad y puntuación F1, lo cual puede indicar que los algoritmos clásicos de aprendizaje automático que se emplearon aquí podrían resultar como opciones oportunas para una tarea de clasificación multietiqueta en un conjunto de datos pequeño, en contraposición a los modelos ligeros basados en transformers.

Dentro de los modelos transformers evaluados, el `distilbert` mostró el desempeño ligeramente más sólido tanto en el entrenamiento con 5 epochs como con 10 epochs. Dicho modelo muestra una precisión y una puntuación F1 más alta en contraste con los otros modelos evaluados.

Tabla 3. Resultados para los modelos SVM y `distilbert-base-multilingual-cased` donde se evalúa si al menos una etiqueta fue predicha correctamente por cada texto.

Modelo	Precisión	Exhaustividad	F1
SVM	**0.91**	**0.65**	**0.76**
`distilbert-base- multilingual-cased (10 epochs)`	0.88	**0.65**	0.73

No obstante, el modelo `twitter-xlm-roberta` también demuestra resultados competitivos, con un rendimiento muy próximo al del `distilbert` en todas las métricas de evaluación e, incluso, con valores superiores en la exhaustividad en un entrenamiento basado en 10 epochs. Por su parte, el modelo `distilroberta-base` presenta un rendimiento notablemente inferior en todas las métricas evaluadas.

Estos hallazgos podrían sugerir al `distilbert` como una opción óptima para esta tarea específica de clasificación de textos multietiqueta, aunque también debería considerarse la alternativa de ajustar los parámetros del modelo `twitter-xlm-roberta-base-sentiment`, el cual ha sido entrenado con textos provenientes de Twitter.

Dado el amplio conjunto de etiquetas y la baja representación de algunas de ellas, como pudo verse en las características del dataset en términos de un desbalance de clases, se decidió llevar a cabo un experimento adicional para evaluar el rendimiento de los modelos tras realizar ciertos ajustes en el conjunto de datos. En este experimento, se excluyeron las etiquetas "cultura pop" y "otro", debido a su escasa presencia y su menor relevancia para la CPC. Al mismo tiempo, se eliminaron los textos que quedaron sin etiquetas después de esta exclusión.

Con estos ajustes, el tamaño del dataset disminuyó a 3629 tuits. El objetivo principal de este experimento fue evaluar la capacidad de los modelos para acertar por lo menos una de las etiquetas verdaderas para cada texto, de la lista de 16 etiquetas restantes. Para ello, se seleccionaron los modelos SVM y distilbert, puesto que fueron los que presentaron un mejor rendimiento según las métricas de evaluación utilizadas. La Tabla 3 presenta los resultados de este experimento con base en las mismas métricas de evaluación utilizadas para la primera parte.

De acuerdo con estos resultados, el modelo SVM mantiene el rendimiento más solido para la tarea de clasificación multietiqueta en términos de precisión y puntaje F1. Este modelo alcanzó una precisión destacada de 0.91, es decir, la mayoría de las etiquetas predichas por el modelo fueron correctas en comparación con las etiquetas reales. Además, el F1 de 0.76 indica un buen equilibrio entre la precisión y la exhaustividad, aunque el valor de 0.65 en la exhaustividad se mantiene como el resultado más bajo del modelo.

Por otro lado, el modelo `distilbert-base- multilingual-cased`, entrenado durante 10 epochs, mostró una precisión ligeramente menor (0.88) en comparación con el SVM. Sin embargo, mantuvo la misma exhaustividad de 0.65, de modo que capturó correctamente el mismo porcentaje de etiquetas reales que el SVM. El F1-score para este modelo fue 0.73, bastante cercano al rendimiento del SVM. En general, los resultados de este experimento suponen una leve mejora en la evaluación de la clasificación multietiqueta para la tarea modificada, en función de las etiquetas más relevantes del corpus.

En todos los casos, tanto en la clasificación general como en el experimento con ajustes de clasificación, se observa que la exhaustividad se mantiene como el valor más bajo entre las métricas de evaluación, derivado de la falta de equilibrio entre las clases. Esto podría indicar que los modelos tienen dificultades para recuperar todos los casos positivos en relación con las etiquetas correctas.

Un nivel bajo de exhaustividad significa que el modelo está dejando pasar cierta cantidad de casos positivos no identificados. En el contexto de la clasificación multietiqueta, una baja exhaustividad puede surgir por varias razones, como la complejidad de las relaciones entre las etiquetas, el desbalance de las clases, la calidad del conjunto de datos utilizado para el entrenamiento o la configuración del modelo. Una forma de abordar este problema es analizar las métricas por cada etiqueta particular para identificar dónde está funcionando mejor el modelo.

Al hacer esto, sería posible identificar tanto las clases para las cuales el modelo tiene un buen desempeño como las que presentan desafíos. De cualquier forma, la intención de este trabajo, una vez evaluados los algoritmos, ha sido la de presentar un modelo general de clasificación entrenado con datos que representan la realidad de la CPC en el contexto mexicano. Si bien los valores de exhaustividad se mantienen como los más bajos, es posible realizar en el futuro un análisis pormenorizado de cada clase, con el objetivo de identificar las áreas de mejor rendimiento del modelo y, posteriormente, seguir una ruta de acción más especializada, con relación a los datos de entrenamiento.

Entre estas rutas, se sugiere la de buscar más ejemplos de las etiquetas menos representadas para conseguir un modelo con un rendimiento general sólido en la clasificación de las áreas temáticas de la CPC, o bien la de concentrarse sólo en las áreas que presentan un mejor rendimiento, con el fin de perfeccionar un modelo con un mejor desempeño general para menos clases.

5. Conclusiones y trabajo futuro

En este trabajo, se ha llevado a cabo una tarea de clasificación automática de textos en el contexto de la CPC en Twitter, sobre la base de un sistema multietiqueta de las áreas temáticas identificadas en el corpus de la investigación. Para lograr el objetivo de desarrollar y evaluar un modelo de clasificación automática, se adoptó un enfoque de aprendizaje automático supervisado, a partir de un corpus anotado, conformado por tuits de CPC publicados en México. En la anotación de este corpus se buscó identificar y clasificar las áreas temáticas presentes en los tuits del corpus, con base en un sistema multietiqueta, donde cada tuit podía podía ser etiquetado con una o más de las 18 categorías temáticas predefinidas.

Con base en este corpus anotado, se evaluaron varios modelos de aprendizaje automático, tanto algoritmos clásicos (SVM y RFC) como modelos basados en arquitecturas transformers (`distilbert-base-multilingual-cased`, `distilroberta-base` y `twitter-xlm-roberta-base-sentiment`). Los resultados de la evaluación, en función de las métricas de precisión, exhaustividad y puntuación F1, destacaron el rendimiento del modelo SVM para predecir correctamente las etiquetas asociadas con los textos del corpus.

Asimismo, los modelos ligeros basados en transformers también ofrecieron resultados competitivos, aunque con ciertas variaciones en las métricas de evaluación. De acuerdo con los resultados reportados en este trabajo, se ha resaltado la necesidad de mejorar la exhaustividad de los modelos en general, especialmente en un contexto de clasificación multietiqueta donde algunas clases tienen una representación limitada en el conjunto de datos.

En este sentido, uno de los desafíos consiste en identificar estrategias efectivas para capturar adecuadamente todas las clases durante el entrenamiento del modelo. Como se ha señalado, una de las limitantes con respecto al tipo de datos del corpus se refiere a la restricción impuesta para recopilar datos de Twitter/𝕏 mediante la API, como consecuencia de los cambios efectuados en la plataforma durante el último año. Para futuros trabajos, se propone investigar enfoques avanzados de procesamiento de texto y aprendizaje automático que puedan mejorar el rendimiento en tareas de clasificación multietiqueta en tuits de CPC.

Esto podría incluir el uso de modelos más grandes o las versiones base de los modelos abordados en este trabajo, así como el ajuste de los hiperparámetros para capturar relaciones complejas entre etiquetas y características del texto. Además, podrían realizarse análisis detallados por cada etiqueta del corpus para identificar las clases con mejores resultados desarrollar estrategias de mejora para las clases con valores de evaluación más bajos. En todo caso, se debe considerar que este estudio representa una de las primeras aproximaciones desde el PLN y el aprendizaje automático a la clasificación de tuits relacionados con la CPC en México.

Los resultados de este trabajo deben intepretarse como una exploración inicial para la clasificación multietiqueta de texto en tuits de CPC. A pesar de que los modelos basados en arquitecturas transformers constituyen un avance significativo en PLN, los algoritmos tradicionales como el SVM siguen mostrando un rendimiento competitivo en la tarea de clasificación multietiqueta automática en datasets pequeños con clases múltiples desbalanceadas. Al abordar la clasificación de tuits de CPC mediante distintos modelos de aprendizaje automático y de aprendizaje profundo, este trabajo establece importantes fundamentos para futuras investigaciones en esta área emergente.

Referencias

1. Aguilar-Tello, V., Angulo-Giraldo, M.: La divulgación científica en twitter durante la pandemia por la COVID 19. Revista Aportes de la Comunicación y la Cultura, vol. 32, no. 32 (2022)
2. Barajas-Galindo, D. E., Rodríguez-Carnero, M. G.: La divulgación científica en los tiempos de twitter. Endocrinología, Diabetes y Nutrición, vol. 67, no. 5, pp. 295–296 (2020) doi: 10.1016/j.endinu.2020.03.001
3. Barbieri, F., Espinosa-Anke, L., Camacho-Collados, J.: XLM-T: Multilingual language models in twitter for sentiment analysis and beyond. In: Proceedings of the 13th Language Resources and Evaluation Conference, European Language Resources Association, pp. 258–266 (2022)
4. Cheplygina, V., Hermans, F., Albers, C., Bielczyk, N., Smeets, I.: Ten simple rules for getting started on twitter as a scientist. PLoS Computational Biology, vol. 16, no. 2, pp. e1007513 (2020) doi: 10.1371/journal.pcbi.1007513

5. Claassen, G.: The viral spreading of pseudoscientific and quackery health messages on twitter - finding a communication vaccine. Current Allergy and Clinical Immunology, vol. 34, no. 1, pp. 18–22 (2021) doi: 10.10520/ejc-caci-v34-n1-a4
6. Daneshjou, R., Shmuylovich, L., Grada, A., Horsley, V.: Research techniques made simple: Scientific communication using twitter. Journal of Investigative Dermatology, vol. 141, no. 7, pp. 1615–1621 (2021) doi: 10.1016/j.jid.2021.03.026
7. Denia, E.: The impact of science communication on twitter: The case of Neil deGrasse Tyson. Comunicar, vol. 28, no. 65, pp. 21–30 (2020) doi: 10.3916/C65-2020-02
8. Denia, E.: Twitter como objeto de investigación en comunicación de la ciencia. Revista Mediterránea de Comunicación, vol. 12, no. 1, pp. 289 (2021) doi: 10.14198/MEDCOM000006
9. Déchène, M., Lesperance, K., Ziernwald, L., Holzberger, D.: From research to retweets—exploring the role of educational twitter (X) communities in promoting science communication and evidence-based teaching. Education Sciences, vol. 14, no. 2, pp. 196 (2024) doi: 10.3390/educsci14020196
10. Guenther, L., Wilhelm, C., Oschatz, C., Brück, J.: Science communication on twitter: Measuring indicators of engagement and their links to user interaction in communication scholars' tweet content. Public Understanding of Science, vol. 32, no. 7, pp. 860–869 (2023) doi: 10.1177/09636625231166552
11. Kowsari, K., Jafari-Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., Brown, D.: Text classification algorithms: A survey. Information, vol. 10, no. 4, pp. 150 (2019) doi: 10.3390/info10040150
12. Lewenstein, B. V.: Models of public communication of science and technology (2003) hdl.handle.net/1813/58743
13. Milbourne, S.: How to use twitter as a scientist (2022) www.letpub.com/How-to-Use-Twitter-as-a-Scientist
14. Peters, H. P., Dunwoody, S., Allgaier, J., Lo, Y. Y., Brossard, D.: Public communication of science 2.0. EMBO Reports, vol. 15, no. 7, pp. 749–753 (2014) doi: 10.15252/embr.201438979
15. Riekert, M., Riekert, M., Klein, A.: Simple baseline machine learning text classifiers for small datasets. SN Computer Science, vol. 2, no. 3, pp. 178 (2021) doi: 10.1007/s42979-021-00480-4
16. Sanchez-Mora, M. C.: Hacia una taxonomía de las actividades de comunicación pública de la ciencia. Journal of Science Communication, pp. 1–9 (2016) ru.ameyalli.dgdc.unam.mx/handle/123456789/73
17. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In: The 5th EMC2 - Energy Efficient Machine Learning and Cognitive Computing Co-located with the 33rd Conference on Neural Information Processing Systems, pp. 1–5 (2020) doi: 10.48550/arXiv.1910.01108
18. Statista Research Department: Countries with most x/twitter users 2023 (2023) www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/
19. Sundström, G.: Science communication on twitter an analysis of vocabulary and content (2021) www.diva-portal.org/smash/get/diva2:1603996/FULLTEXT01.pdf
20. Zappavigna, M.: The discourse of Twitter and social media. Continuum International Publishing Group (2012) doi: 10.5040/9781472541642

ISSN 1870-4069

Aproximación de señales ECG y EEG mediante redes neuronales de pulso

Omar Samperio-Vázquez[1], Juan Carlos González-Islas[1], Luis Enrique Ramos-Velasco[2], Jesus Patricio Ordaz-Oliver[1], Gildardo Godinez-Garrido [1,3]

Universidad Autónoma del Estado de Hidalgo,
Hidalgo,
México

Universidad Metropolitana del Estado de Hidalgo,
Hidalgo,
México

Universidad Tecnológica de Tulancingo,
Hidalgo,
México

{omarsamvaz, juan_gonzalez7024, jesus_ordaz}@uaeh.edu.mx,
lramos@upmh.edu.mx, gildardo.godinez@utectulancingo.edu.mx

Resumen. La aproximación de señales, como el electrocardiograma,el electroencefalograma y el electromiograma (ECG,EEG, por sus siglas en inglés, Electrocardiography, Electroencephalography) mediante modelos matemáticos o computacionales es un área de interés para aplicaciones médicas. En este trabajo de investigación se propone una arquitectura de redes neuronales de pulso para la aproximación de señales fisiológicas electroencefalográficas y electrocardiográficas. En la arquitectura se emplea una señal de persistencia con la finalidad de evitar singularidades del algoritmo de entrenamiento. El esquema propuesto puede ser generalizado y aplicado para aproximar otro tipo de señales tanto fisiológicas como biomecánicas del cuerpo humano con fines de diagnóstico o evaluación de anomalías.

Palabras clave: Redes neuronales de pulso, señales fisiológicas, ECG, EEG.

Approximation of ECG and EEG Signals through Spiking Neural Networks

Abstract. The approximation of signals, such as ECG, and EEG (Electrocardiography, Electroencephalography) using mathematical or computational models is an area of interest for medical applications. In this research work, an architecture of pulse neural networks is proposed for the approximation of electroencephalographic and electrocardiographic physiological signals. In the architecture, a persistence signal is used in order to avoid singularities in the training algorithm. The proposed scheme can

be generalized and applied to approximate other types of physiological and biomechanical signals from the human body for the purposes of diagnosis or evaluation of anomalies.

Keywords: Spiking neural networks, physiological signals, ECG, EEG.

1. Introducción

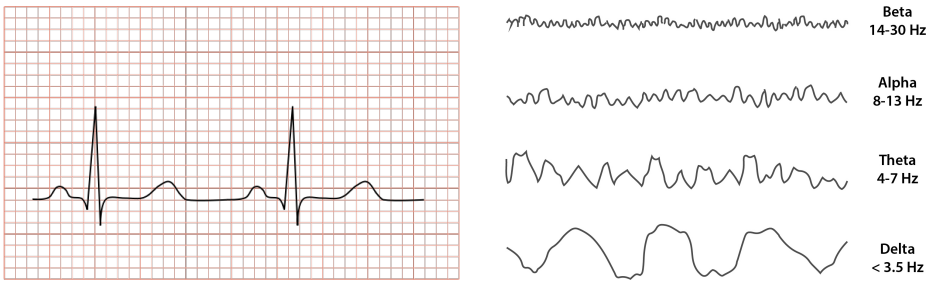
El modelado matemático y la simulación computacional de señales fisiológicas del cuerpo humano como lo son: el electrocardiograma, el electroencefalograma y el electromiograma (ECG,EEG y EMG, por sus siglas en inglés, Electrocardiography, Electroencephalography, y Electromiography respectivamente), representan un campo de investigación emergente y de sumo interés. Dichas señales fisiológicas son utilizadas por los médicos para diagnosticar el comportamiento normal o irregular de los órganos humanos como el corazón y el cerebro, entre otros [14, 13, 24]. La generación de señales ECG, es ampliamante usado para la prueba, calibración y mantenimiento de equipo de electrocardiografía.

Las señales generadas son mayormente señales ideales y generalmente no se aproximan a las señales reales que contienen ruido [3, 23]. Existen diferentes métodos para generar señales ECG como los basados en: derivadas fraccionarias [6], ecuaciones dinámicas [4], ecuaciones estáticas [15], polinomios de Chebyshev [5, 26]; así como, señales reales almacenadas en bases de datos de ECG [11].

De igual manera, se han reconstruido señales ECG a partir de la adquisición con la plataforma Shimmer y los algoritmos de Emparejamiento orthogonal (OMP por sus siglas en inglés, Orthogonal Matching Pursuit) [12]; o con el uso de un filtro de media móvil y la eliminación de los cruces por cero [25]. En cuanto a las señales de EEG son útiles para monitorear las actividades cerebrales para tareas médicas (detección de convulsiones) y cognitivas (reconocimiento de emociones, interfaz cerebro-computadora). Por lo que el sensado, procesamiento y reconstrucción son de sumo interés para la comunidad científica [20, 22].

Para reconstruir muestras de las señales EEG se han usado redes Neuronales Artificiales de segunda generación (ANN por sus siglas en inglés, Artifitial Neural Networks) utilizando subconjuntos de registros de señales pregrabadas [17]. Las redes neuronales de pulso también llamadas redes neuronales de tercera generación (SNN por sus siglas en inglés, Spikin Neural Networks) representan una clase especial de las redes neuronales artificiales, donde los modelos de neuronas se comunican mediante secuencias de pulsos como lo hacen los órganos biológicos [10].

Las redes compuestas de neuronas con pulsos son capaces de procesar una cantidad sustancial de datos utilizando un número relativamente pequeño de pulsos, lo que reduce el tiempo de procesamiento y el consumo de energía [27]. Las SNN resuelven una variedad de problemas específicos en ingeniería aplicada, como el procesamiento rápido de señales, la detección de eventos, la clasificación, el reconocimiento de voz, la navegación espacial o el control de motores. Se ha demostrado que las SNN se puede aplicar no sólo a todos los problemas que pueden resolverse mediante redes neuronales



(a) Señal de electrocardiograma normal [1].

(b) Señales de encefalograma (dispositivo Emotiv Epoc).

Fig. 1. Señales fisiológicas ECG y EEG.

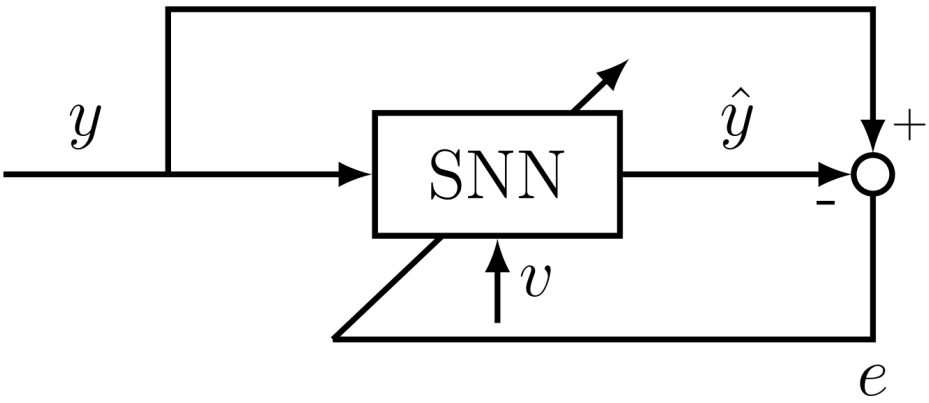


Fig. 2. Esquema propuesto para la aproximación de señales ECG y EEG.

artificiales de segunda generación, sino que los modelos con pulsos son de hecho computacionalmente más poderosos que los perceptrones y las compuertas sigmoidales [19]. Enfoques recientes, han usado SNNs para aproximar de manera eficiente señales electrofisiológicas grabadas, usando una estrategia de cómputo evolutivo y programación de expresión de genes [8, 7]. Los principales desafíos en la generación, síntesis, aproximación o reconstrucción de señales fisiológicas, se centra en tareas de detección, sensado, procesamiento y almacenamiento, con un error mínimo y bajo consumo de energía.

Por tanto, en este trabajo de investigación se propone una arquitectura de redes neuronales de pulso, para la aproximación de señales médicas electroencefalográficas y electrocardiográficas. La organización del resto del artículo está dada de la siguiente manera: en la Sección 2 se presentan el esquema propuesto para la aproximación de las señales biológicas y los diagramas de las SNN, mientras que en la Sección 3 se dan los resultados obtenidos de la aproximación de las señales ECG y EEG, finalmente las conclusiones se enuncian en la Sección 4.

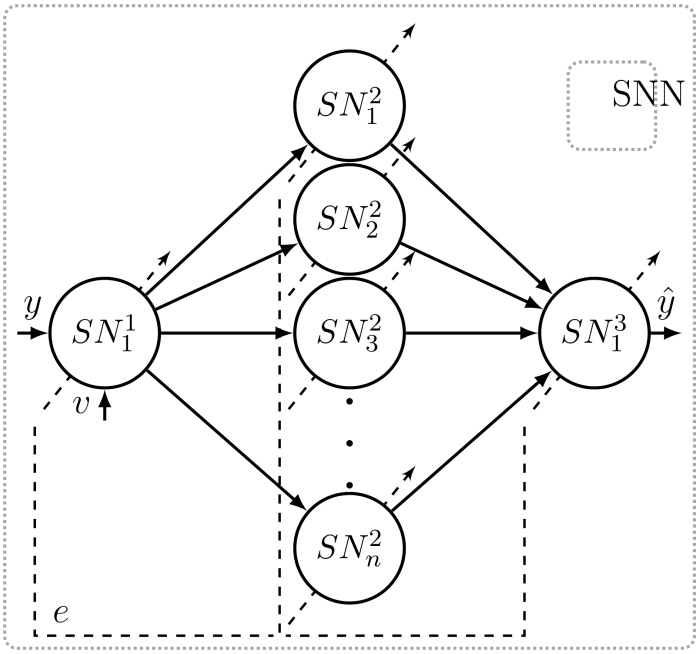


Fig. 3. Diagrama General de una SNN con tres capas.

2. Materiales y métodos

El electrocardiograma (Fig. 1a) y el electroencefalograma (Fig. 1b), registran las señales eléctricas del corazón y el cerebro, respectivamente. En el primer caso se colocan electrodos en el pecho para registrar las señales eléctricas que originan los latidos, mientras que para las EEG se miden en el cuero cabelludo mediante electrodos pasivos [14]. En este trabajo para el ECG se utiliza una señal de un adulto con características normales1a, para el caso del EEG se utiliza la actividad de imaginación de movimiento obteniendo la señal de uno de los ocho electrodos del dispositivo Emotiv Epoc colocados: AF3, AF4, FC5, FC6, P7, P8, O1 y O2, siendo los datos del electrodo AF4.

2.1. Esquema propuesto

Existen diversos tipos de modelos matemáticos que representan la dinámica de las neuronas, los cuales son categorizados con base en su nivel de abstracción [9]. En este trabajo se hace uso del modelo matemático de las neuronas con integración y disparo perfecto (PIF por sus siglas en inglés, Perfect integration) [9], con entrenamiento supervisado SpikeProp [2]. En la Fig. 2 se muestra el diagrama a bloques del esquema propuesto. Donde, y es la señal a ser aproximada mediante la red neuronal tipo pulso y el entrenamiento está supeditado a la señal del error e que se genera de la diferencia entre la señal a aproximar y y la señal resultante $\hat{y}$. Para el caso de estudio, la entrada y corresponde a las señales EEG o ECG, respectivamente.

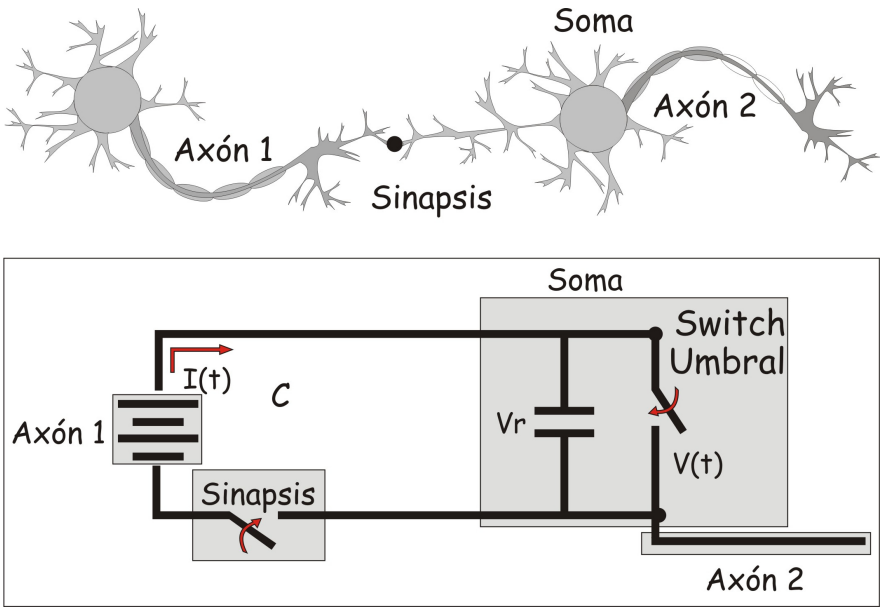


Fig. 4. Representación del modelo integración y disparo perfecto (PIF), de una neurona.

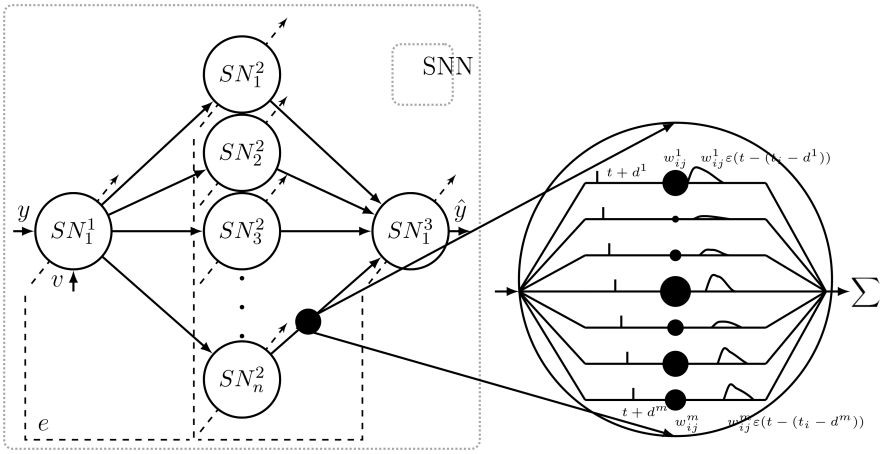
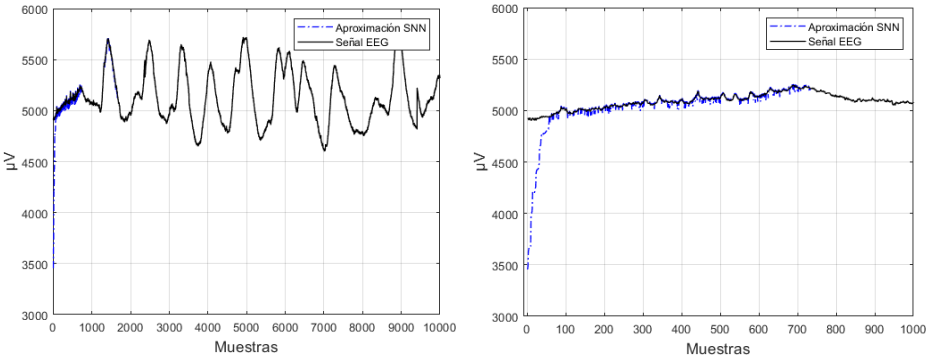


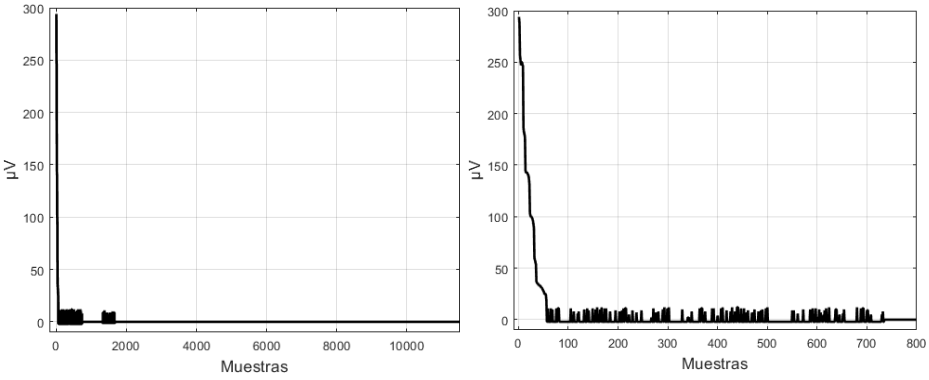
Fig. 5. Representación de los pesos w en el entrenamiento de una SNN [21].

La salida de la SNN $\hat{y}$ es la aproximación de la señal deseada, v es una señal de excitación persistente a la neurona SN_1^1. Para estos experimentos se considera un pulso equivalente a un disparo neuronal en el instante 200, dentro en la ventana de tiempo de 0 a 1000 muestras, con la finalidad de evitar singularidades del algoritmo. La arquitectura propuesta para este trabajo se muestra en la Fig. 3, dado que se está trabajando con solo una señal de entrada a la vez por tanto se requiere una neurona en la capa de entrada y otra neurona en la capa de salida, es decir, la capa de entrada tiene una neurona SN_1^1, una capa oculta SN_n^2 con n-neuronas y una capa de salida SN_1^3 con una neurona.



(a) (b)

Fig. 6. Aproximación de SNN de la señal EEG, con 5 neuronas en la capa oculta.



(a) (b)

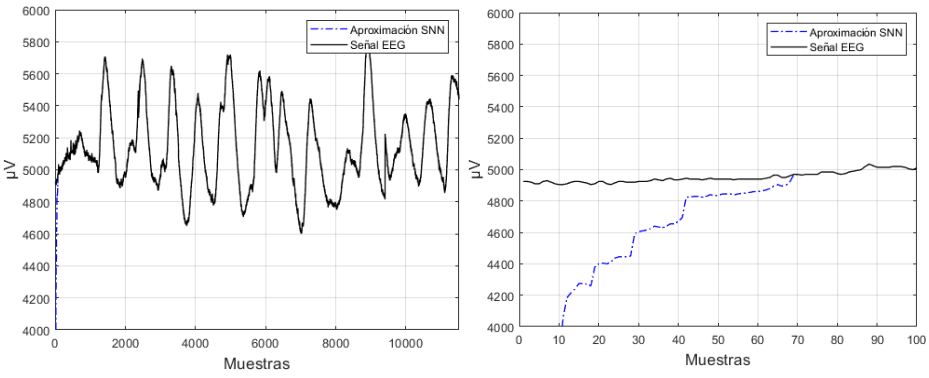
Fig. 7. Error de aproximación de la señal EEG, con 5 neuronas en la capa oculta.

El número de neuronas n en la capa oculta depende de la complejidad de la señal de referencia y su selección se hace de manera aleatoria tomando en cuenta la experiencia del usuario experto que programa el algoritmo, por lo que, en este caso, de manera inicial se proponen dos valores $n = 5$ para la señal EEG y $n = 10$ para la señal ECG. El valor de n se modifica hasta encontrar la mejor aproximación dependiendo el error. Cabe hacer mención que también se propusieron valores para n más grandes como $n = 20$, $n = 50$, sin embargo, no se encontraron mejores resultados. Dado, que la red neuronal se entrena con el error, por tanto, para generarlo se requieren la señales y y $\hat{y}$ codificada, con esto se obtiene la diferencia entre los tiempos de disparo específicos correspondientes a salidas deseadas y las entradas a la red neuronal. La función de error está dada por:

$$e = y - \hat{y}. \tag{1}$$

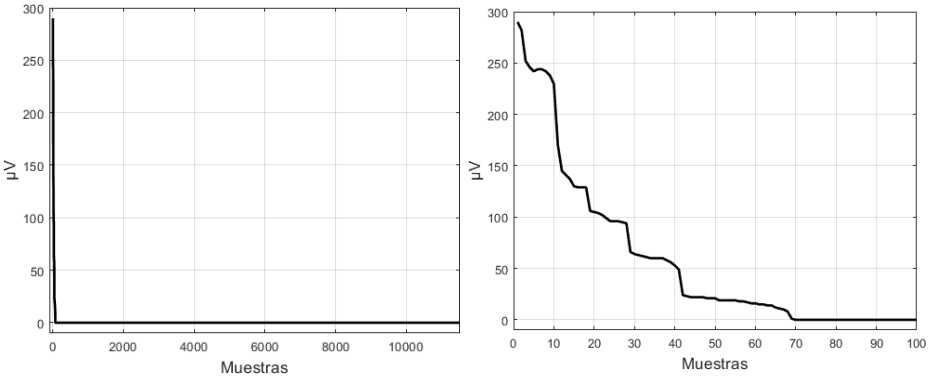
2.2. La Neurona tipo integración y disparo perfecto

Con la finalidad de comprender el modelo de neurona de integración y disparo perfecto PIF, se emplea un circuito capacitivo como se ilustra en la Fig. 4, donde el



(a) (b)

Fig. 8. Aproximación de SNN de la señal EEG, con 10 neuronas en la capa oculta.



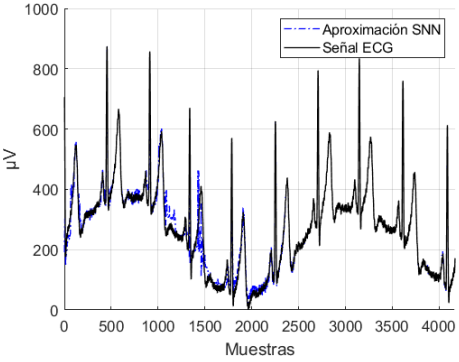
(a) (b)

Fig. 9. Error de aproximación de la señal EEG, con 10 neuronas en la capa oculta.

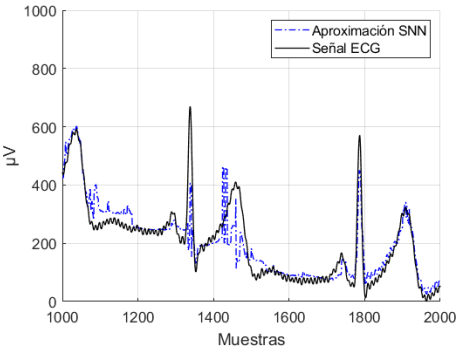
voltaje de entrada se va acumulando en el capacitor y es disparado cuando se cierra el circuito, dicho disparo se presenta en una neurona biológica al momento de hacer la sinapsis. La ecuación diferencial que describe este sistema de neurona está dada por:

$$V(t) = V_r + \frac{1}{C}\int_{t_0}^{t} I(t)\,dt, \tag{2}$$

donde $V(t)$ es el voltaje acumulado en el capacitor, V_r es el valor del voltaje en reposo del capacitor, o voltaje de relajación de la neurona, propuesto de $-65\,\mathrm{mV}$, C es el valor del capacitor en µF propuesto de 50, e $I(t)$ es el valor de la corriente de la fuente o señal de entrada. Para el caso de estudio, se eligió una neurona del tipo integración y disparo perfecto PIF considerándola una de las más sencillas al momento de programarla y que no requiere un costo computacional alto.

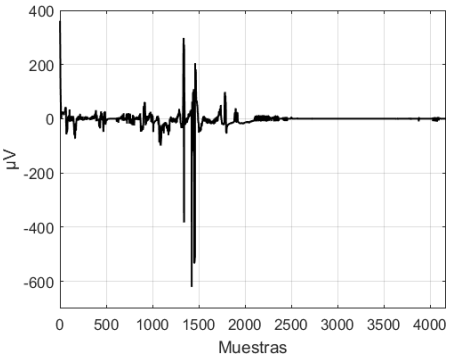


(a)

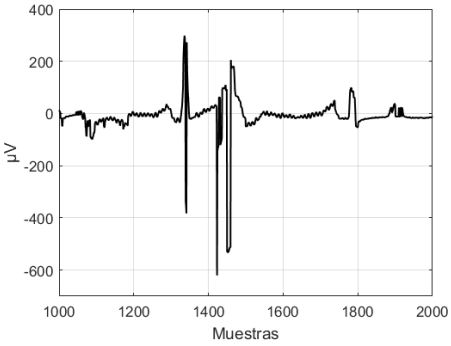


(a)

Fig. 10. Aproximación de SNN de la señal ECG, con 5 neuronas en la capa oculta.



(a)



(b)

Fig. 11. Error de aproximación de la señal ECG, con 5 neuronas en la capa oculta.

2.3. Codificación

El arreglo de neurona en desarrollo requiere un esquema de codificación de velocidad o de tiempo. Generalmente, la primero se hace utilizando la información de la velocidad de disparo (determinado por el número de spikes), mientras que la segunda emplea el disparo como información de tiempo para traducir un patrón de pulso [16]. En este trabajo se usa la codificación temporal, esto es, en la neurona se ajustan los pesos w de manera que durante la ventana de tiempo de 1000 épocas se obtiene un disparo de la neurona spike.

2.4. Método de aprendizaje spikeprop

Bohte et al. [18] propusieron un algoritmo de aprendizaje supervisado basado en el algoritmo BackPropagation de redes neuronales artificiales de segunda generación. Este método se denomina SpikeProp, y fue diseñado para una arquitectura de redes neuronales con múltiples conexiones las cuales tiene múltiples retardos.

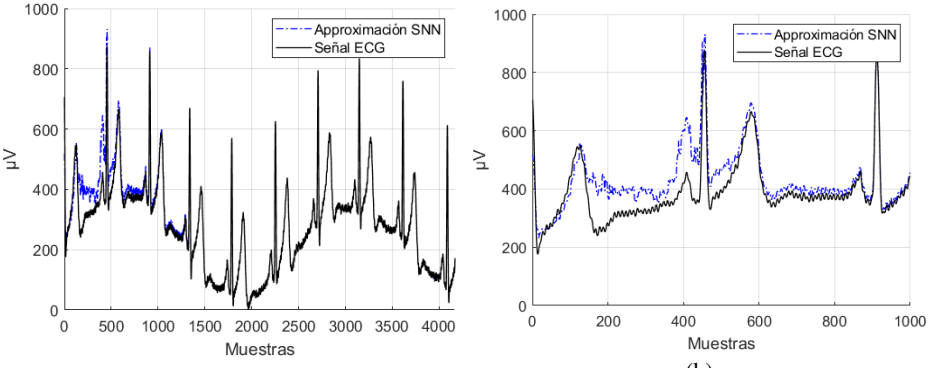


Fig. 12. Aproximación de SNN de la señal ECG, con 10 neuronas en la capa oculta.

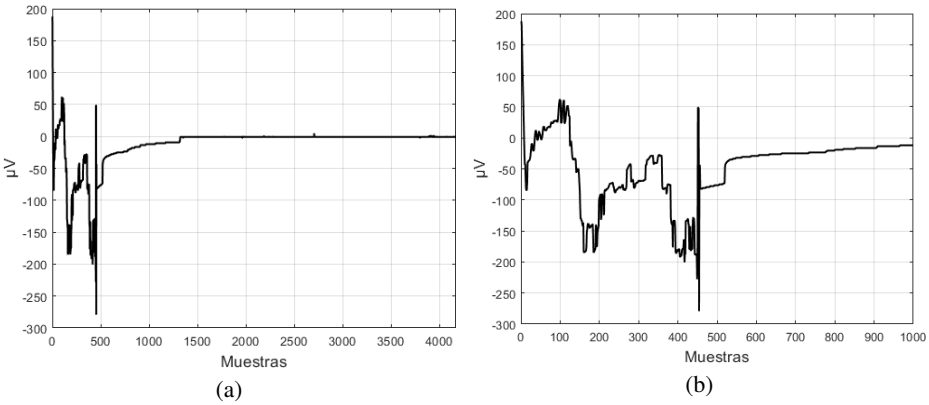


Fig. 13. Error de aproximación de la señal ECG, con 10 neuronas en la capa oculta.

La arquitectura que necesita este entrenamiento es de tipo feedforward, lo que significa que no existe retroalimentación entre ninguna neurona de la red, aunque esto no limita a que la red neuronal tenga una o más capas ocultas. La adaptación de los pesos neuronales relacionados a la neurona de la capa de salida es:

$$\Delta\omega_{ij}^{k} = -\eta\frac{\partial e}{\partial\omega_{ij}^{k}}, \tag{3}$$

donde e es el error y se obtiene mediante (1), ω_{ij}^{k} son los k-ésimos pesos de la neurona i-ésima a la neurona j-ésima, como se muestra en la Fig. 5, y η es la tasa de aprendizaje. Obteniendo la derivada parcial del error con respecto a los pesos se obtiene:

$$\Delta\omega_{ij}^{k} = -\eta\frac{y_i^k(t_j)e}{\sum\limits_{i,l}\omega_{ij}^{l}(\dfrac{\partial y_j^l(t_j)}{\partial t_i})}, \tag{4}$$

donde $\Delta\omega_{ij}^{k}$ es la modificación que sufren los pesos para llegar a la obtención del peso ideal, tal que la salida $\hat{y}$ sea la que sea la señal aproximada con el mínimo error.

En (3) se tiene el término llamado tasa de aprendizaje η, que en este trabajo se propuso de 1.5, cabe hacer mención que este parámetro se obtiene de forma empírica proponiendo diferentes valores hasta encontrar el mejor desempeño de la red, para que el tiempo de convergencia sea mínimo.

3. Resultados

3.1. Experimento con EEG

En este experimento se toma una señal EEG obtenida con el dispositivo Emotiv Epoc para ser aproximada con una SNN, en este caso se usó la señal de imaginación de movimiento. Empleando dos arquitecturas diferentes con 5 y con 10 neuronas en la capa oculta, respectivamente este número de neuronas se propuso de manera empírica mencionando que se realizaron experimento con menos de 5 neuronas y con más de 10 neuronas, sin embargo, los resultados no son mejores que los mostrados en este trabajo, también se propuso una tasa de aprendizaje $\eta = 1.5$ en las arquitecturas, observando las diferencias entre ambas.

En la Fig. 6 se observa la aproximación de la señal EEG mediante una SNN, con una arquitectura de 5 neuronas en la capa oculta, así como un acercamiento entre 0 y 1000 muestras para denotar la aproximación antes de la disminución del error. En la Fig. 7, se muestra el error para la aproximación de la señal EEG, con 5 neuronas y un acercamiento igual que en la Fig. 6. Como puede observarse, con 5 neuronas, a partir de la muestra 80 el error se aproxima a 0, sin embargo, hay oscilaciones y es hasta después de la muestra 1500, que el error tiende a 0. Cuando se aumenta el número de neuronas en la capa oculta, se mejora la capacidad de aproximación, aunque existe un compromiso con el costo computacional.

En la Fig. 8 se presentan los resultados de la aproximación y un acercamiento de la señal EEG mediante una SNN, empleando 10 neuronas en la capa oculta. De igual manera que en el caso anterior con una SNN de 5 neuronas, la evaluación de la aproximación se hace mediante el error. En la Fig. 9. Como puede observarse en la Fig. 8 y la Fig. 9 la señal EEG aproximada con una SNN con 10 neuronas en la capa oculta tiene un tiempo de convergencia menor. El error se aproxima a cero en en lapso menor a las 100 muestras (lo que representa un intervalo menor a la frecuencia de muestreo del sensor Epoc emotiv).

3.2. Experimento con ECG

Para este experimento se utiliza una arquitectura de una neurona en la capa de entrada, 5 y 10 neuronas en la capa oculta y una neurona en la capa de salida, así como una tasa de aprendizaje $\eta = 1.5$ en ambas arquitecturas. De manera similar al experimento anterior, la Fig. 10 presenta la señal aproximada EEG y un acercamiento mediante una SNN con 5 neuronas en la capa oculta. Por su parte, la Fig. 11 muestra el error de aproximación para este caso. En el experimento con la señal ECG se observa que esta señal es más compleja en su composición, sin embargo, la SNN la aproxima de forma precisa, y con la red de 5 neuronas en la capa oculta, el error lo acerca a cero

a partir de las 2500 muestras. Bajo la misma suposición que en el experimento anterior se mejora la aproximación, en las Figs. 12 y 13, se muestra la señal aproximada ECG y el error de aproximación para una SNN de con 10 neuronas en la capa oculta.

Con la arquitectura de 10 neuronas en la capa oculta el error lo mantiene acotado cerca del cero a partir de las 1500 muestras como se observa en la Fig. 13, mientras que con 5 neuronas el error es más amplio y el tiempo de convergencia es mayor. Una prueba estadística de normalidad del error, que se aplicó a las señales de error de los experimentos es la prueba Kolmogorov-Smirnov, y como resultado muestra que no hay errores sistemáticos, lo que significa que el error tiene una distribución normal estándar (media cero y desviación estándar igual a uno) con un nivel de significación del 5 %.

4. Conclusiones

De acuerdo a los resultados obtenidos se puede concluir que, las redes neuronales de pulso son útiles para aproximar con precisión y rapidez señales ECG y EEG, ya que requieren un número menor de neuronas. La arquitectura de red propuesta compuesta de tres capas, con una neurona tanto en la capa de entrada como en la capa de salida. Debido que no existe un método para determinar la cantidad de neuronas necesarias en la capa oculta, se determinó heurísticamente empleando 5 y 10 neuronas, respectivamente. Si bien se propusieron valores de n mayores a 10, los mejores resultados se obtuvieron con la arquitectura de 10 neuronas, tanto para las señales ECG como para las señales EMG. El error converge a cero en menos de 100 muestras de un total de 11,520 para el caso de la EEG. y para el caso de la ECG la convergencia a cero la realiza en las 1000 muestras de un total de 4120. El esquema propuesto se puede ampliar para aproximar otras señales de menor o mayor complejidad, como señales electromiográficas o inherentes a la respiración o el ciclo de marcha humano, con fines de diagnóstico clínico.

Referencias

1. Azcona, L.: El electrocardiograma. Libro de la salud cardiovascular del Hospital Clínico San Carlos y la fundación BBVA, pp. 49–56 (2009)
2. Bohte, S. M., Kok, J. N., La-Poutré, H.: Error-backpropagation in temporally encoded networks of spiking neurons. Neurocomputing, vol. 48, no. 1–4, pp. 17–37 (2002) doi: 10.1016/s0925-2312(01)00658-0
3. Ciucu, R. I., Seriţan, G. C., Dragomir, D. A., Cepişcă, C., Adochiei, F. C.: ECG generation methods for testing and maintenance of cardiac monitors. In: E-Health and Bioengineering Conference, pp. 1–4 (2015) doi: 10.1109/EHB.2015.7391513
4. Clifford, G. D., McSharry, P. E.: A realistic coupled nonlinear artificial ECG, BP, and respiratory signal generator for assessing noise performance of biomedical signal processing algorithms. Fluctuations and Noise in Biological, Biophysical, and Biomedical Systems II, vol. 5467, pp. 290–301 (2004) doi: 10.1117/12.544525
5. Daoui, A., Yamni, M., Karmouni, H., Sayyouri, M., Qjidaa, H.: Efficient reconstruction and compression of large size ECG signal by Tchebichef moments. In: International Conference on Intelligent Systems and Computer Vision, pp. 1–6 (2020) doi: 10.1109/ISCV49265.2020.9204132

6. Das, S., Maharatna, K.: Fractional dynamical model for the generation of ECG like signals from filtered coupled Van-der Pol oscillators. Computer Methods and Programs in Biomedicine, vol. 112, no. 3, pp. 490–507 (2013) doi: 10.1016/j.cmpb.2013.08.012
7. Espinosa-Ramos, J. I., Cortés, N. C., Vázquez, R. A.: Spiking neuron model approximation using GEP. In: IEEE Congress on Evolutionary Computation, pp. 3260–3267 (2013) doi: 10.1109/CEC.2013.6557969
8. Espinosa-Ramos, J. I., Vazquez, R. A., Cruz-Cortes, N.: Designing spiking neural models of neurophysiological recordings using gene expression programming. BMC Neuroscience, vol. 14, no. 1, pp. P74 (2013) doi: 10.1186/1471-2202-14-S1-P74
9. Gerstner, W., Kistler, W. M.: Spiking neuron models: Single neurons, populations, plasticity. Cambridge University Press (2002) doi: 10.1017/cbo9780511815706
10. Iannella, N., Back, A. D.: A spiking neural network architecture for nonlinear function approximation. Neural Networks, vol. 14, no. 6–7, pp. 933–939 (2001) doi: 10.1016/s0893-6080(01)00080-6
11. Kaur, G.: Design and development of dual channel ECG simulator and peak detector. Master's Thesis, Thapar Institute of Engineering and Technology Deemed University (2006)
12. Kerdjidj, O., Ghanem, K., Amira, A., Harizi, F., Chouireb, F.: Real ECG signal acquisition with shimmer platform and using of compressed sensing techniques in the offline signal reconstruction. In: Proceedings of the IEEE International Symposium on Antennas and Propagation, pp. 1179–1180 (2016) doi: 10.1109/APS.2016.7696297
13. Klabunde, R.: Cardiovascular physiology concepts. Lippincott Williams and Wilkins (2011)
14. Klem, G. H., Lüders, H. O., Jasper, H. H., Elger, C.: The ten-twenty electrode system of the international federation. Recommendations for the Practice of Clinical Neurophysiology: Guidelines of the International Federation of Clinical Physiology EEG Suppl 52, vol. 52, pp. 3–6 (1999)
15. Kovacs, P.: ECG signal generator based on geometrical features. In: Annales Universitatis Scientiarum Budapestinensis de Rolando Eotvos Nominatae, Sectio Geologica, vol. 37, pp. 247–260 (2012)
16. Liu, J., Hu, Y., Li, G., Pei, J., Deng, L.: Spike attention coding for spiking neural networks. IEEE Transactions on Neural Networks and Learning Systems, pp. 1–7 (2023) doi: 10.1109/TNNLS.2023.3310263
17. McBride, J., Zhao, X., Munro, N., Jiang, Y., Smith, C., Jicha, G.: Scalp EEG signal reconstruction for detection of mild cognitive impairment and early Alzheimer's disease. In: Biomedical Sciences and Engineering Conference, pp. 1–4 (2013) doi: 10.1109/BSEC.2013.6618497
18. McKennoch, S., Liu, D., Bushnell, L. G.: Fast modifications of the spikeprop algorithm. In: Proceedings of the IEEE International Joint Conference on Neural Network Proceedings, pp. 3970–3977 (2006) doi: 10.1109/IJCNN.2006.246918
19. Ponulak, F., Kasinski, A.: Introduction to spiking neural networks: Information processing, learning and applications. Acta neurobiologiae experimentalis, vol. 71, no. 4, pp. 409–433 (2011) doi: 10.55782/ane-2011-1862
20. Ramakrishnan, A. G., Satyanarayana, J. V.: Reconstruction of EEG from limited channel acquisition using estimated signal correlation. Biomedical Signal Processing and Control, vol. 27, pp. 164–173 (2016) doi: 10.1016/j.bspc.2016.02.004
21. Samperio-Vázquez, O.: Control de un sistema subactuado empleando redes neuronales de pulso. Master's Thesis, Universidad Autónoma del Estado de Hidalgo (2016)
22. Singh, W., Shukla, A., Deb, S., Majumdar, A.: Energy efficient acquisition and reconstruction of EEG signals. In: Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 1274–1277 (2014) doi: 10.1109/EMBC.2014.6943830

23. Vidhya, V., Unnikrishnan, D.: Synthetic ECG and PPG signal generation using pulse shaping technique. In: Proceedings of the Annual IEEE India Conference, pp. 1–6 (2015) doi: 10.1109/INDICON.2015.7443256
24. Waller, A. D.: A demonstration on man of electromotive changes accompanying the heart's beat. The Journal of Physiology, vol. 8, no. 5, pp. 229–234 (1887) doi: 10.1113/jphysiol.1887.sp000257
25. Wu, C. H., Liu, W. X., Lin, M. S., Chen, J. J.: An ECG extraction and reconstruction system with dynamic EMG filtering implemented on an ARM chip. In: IEEE International Conference of Intelligent Applied Systems on Engineering, pp. 62–65 (2019) doi: 10.1109/iciase45644.2019.9074076
26. Yadav, O. P., Ray, S.: Efficient ECG approximation using Chebyshev polynomials. In: Proceedings of the International Conference on Inventive Research in Computing Applications, pp. 1110–1115 (2018) doi: 10.1109/ICIRCA.2018.8597372
27. Yamazaki, K., Vo-Ho, V. K., Bulsara, D., Le, N.: Spiking neural networks and their applications: A review. Brain Sciences, vol. 12, no. 7, pp. 863 (2022) doi: 10.3390/brainsci12070863

ISSN 1870-4069

Detección de pimiento morrón utilizando TinyML

Jesús A. Martínez-Vargas, Said Polanco-Martagón,
Yahir Hernández-Mier, Marco A. Nuño-Maganda

Universidad Politécnica de Victoria,
Sistemas Inteligentes,
México

{1930337, spolancom, yhernandezm, mnunom}@upv.edu.mx

Resumen. En este trabajo se presenta un sistema TinyML y visión artificial para la detección de pimientos morrones. Se utilizó una red neuronal convolucional con arquitectura MobileNet la cual se implementó en un dispositivo de borde de bajo costo y baja potencia, la ESP32CAM. Para el proceso de entrenamiento se optó por utilizar la técnica de transferencia de conocimiento (transfer learning) de pesos pre entrenados con el conjunto de datos COCO 2017 para disminuir el tiempo de cómputo requerido. Se utilizó el framework Edge Impulse con un conjunto de datos con un total de 1910 imágenes de pimientos morrones y otros objetos que pudiesen encontrarse en un huerto. La exactitud del modelo quantizado a int8 fue de 91.12 % contra el conjunto de prueba. Los resultados obteniendos se muestran favorables y alentadores para trabajos futuros.

Palabras clave: Aprendizaje profundo, tinyML, sistemas embebidos, cómputo de borde.

Detecting Bell Peppers through TinyML

Abstract. In this work, a TinyML and computer vision system for the detection of bell peppers is presented. A convolutional neural network with MobileNet architecture was used, which was implemented on a low-cost and low-power edge device, the ESP32CAM. For the training process, it was chosen to use the transfer learning technique of pre-trained weights with the COCO 2017 dataset to reduce required computation time. The Edge Impulse framework was used with a dataset with a total of 1910 images of bell peppers and other objects that could be found in a chili garden. The accuracy of the model quantized to int8 was 91.12% against the test set. The obtained results are shown favorable and encouraging for future work.

Keywords: Deep learning, tinyML, embedded systems, edge computing.

1. Introducción

En la actualidad, el crecimiento de la población junto con la migración de personas del campo hacia áreas urbanas ha provocado un aumento en la demanda de productos alimenticios. De acuerdo con el Instituto Nacional de Estadística y Geografía (INEGI),

Tabla 1. Estado del arte de sistemas TinyML aplicados a la agricultura.

Ref	Modelo	Dataset	Exactitud	Dispositivo de borde	Objetivo
[6]	CNN + LSTM	Plan Village	97.18 %	Servidor de borde	Predicción de enfermedades de frutas.
[21]	MLP	Propio	99 %	Raspberry Pi 3B	Inspección de la naraja.
[9]	CNN	Propio	97.39 %	ESP32-S3 (MCU)	Proceso post-cosecha de la fruta del olivo.
[13]	MobileNet	VegFru	82.72 %	Teléfono móvil	La clasificación de imágenes de frutas y verduras.
[7]	Inception	DiaMOS	99.73 %	ESP32-CAM	Identificación de enfermedades en la pera.

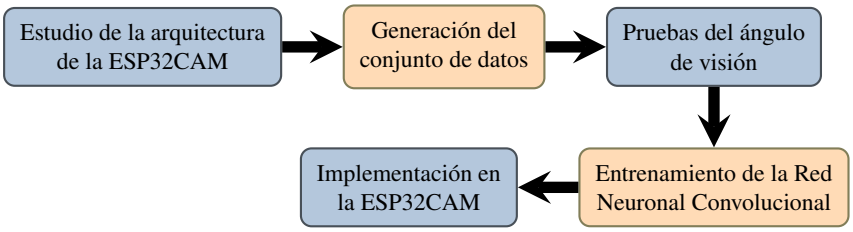


Fig. 1. Etapas de implementación de solución.

en México el porcentaje de población en áreas urbanas creció del 43 % en 1950 al 79 % en el 2020. Por otro lado, en 1950 la población rural constituia el 57 % y para el 2020 se ubicaba en 21 % [11]. Esta migración ha reducido el número de personas que trabajan en las zonas rurales, lo que ha provocado una escasez de mano de obra y una menor producción agrícola, lo que ha propiciado el uso de técnicas de cultivo sin suelo, cómo son la aereoponia y la hidropónia [3].

El uso de dichas técnicas ha facilitado el cultivo en regiones donde las prácticas agrícolas tradicionales son complicadas debido a condiciones climáticas o del suelo dando como resultado una mejora en la cantidad de productos hortícolas, lo que indica un futuro prometedor para la agricultura urbana [5]. Además, estas técnicas de cultivo sin suelo han sido reconocidas por su capacidad para mejorar la eficiencia en el uso de nutrientes y la calidad de los cultivos [24].

La agricultura sin suelo se encamina hacia una automatización integral, abarcando desde la germinación de semillas hasta la cosecha. Esta transformación se impulsa por la aplicación de inteligencia artificial, visión artificial, aprendizaje profundo y sistemas de monitoreo, optimizando cada fase del proceso productivo [22]. En este contexto, TinyML emerge como una tecnología clave para la agricultura de precisión porque ayuda al análisis e inferencias de datos localmente, reduciendo la latencia, mejorando la seguridad y la privacidad, y al ahorro del ancho de banda al evitar la necesidad de enviar datos a servidores centralizados para su procesamiento.

De esta forma, TinyML describe el uso de modelos de aprendizaje automático (ML, por sus siglas en inglés) para dispositivos periféricos o unidades de microcontrolador (MCU) con capacidades computacionales restringidas [17]. Con la ayuda de este enfoque los algoritmos de ML se pueden implementar y entrenar en dispositivos compactos de bajo consumo, lo que permite el procesamiento de datos en tiempo real y la toma de decisiones en el borde de la red. Por lo anterior, en este artículo se presenta un sistema de detección de pimientos morrón enfocado a la agricultura sin suelo, basado en dispositivos de borde y TinyML.

(a) Sub conjunto de pimientos normalizado. (b) Sub conjunto de no-pimiento normalizado.

Fig. 2. Conjunto de datos redimensionado.

(a) Imagen de 96×96 a 18 cm. (b) Imagen de 96×96 a 24 cm. (c) Imagen de 160×120 a 18 cm.

Fig. 3. Capturas realizadas sin flash de la ESP32CAM.

En contraste con los enfoques tradicionales donde se requieren servidores de gran capacidad computacional para ejecutar modelos de aprendizaje profundo complejos, nuestro sistema propone una solución de bajo costo y eficiente. La propuesta consiste en un dispositivo ESP32CAM que implementa una red neuronal MobileNet entrenada para detectar pimientos morrones en imágenes. El entrenamiento se realizó con un conjunto de datos de 1910 imágenes, principalmente obtenidas de internet. Los resultados obtenidos demuestran el potencial de TinyML para contribuir a la seguridad alimentaria, la eficiencia en el uso de recursos y la sostenibilidad de la agricultura en áreas urbanas.

2. Antecedentes

En la literatura existen diversos trabajos donde se hace uso de las redes neuronales y el aprendizaje profundo para el monitoreo del pimiento morron en diversas etapas. Ejemplos de ello son [15, 14] donde utilizan modelos de redes neuronales convolucionales para la identificación de enfermedades de pimiento morrón a través de la clasificación de sus hojas. Para esto, utilizaron una versión reducida el conjunto de datos Plant Village donde sólo se centran en las hojas de pimiento morrón.

(a) Imagen de 160×120 a 24 cm. (b) Imagen de 240×176 a 18 cm. (c) Imagen de 176×144 a 24 cm.

Fig. 4. Capturas realizadas utilizando ESP32CAM.

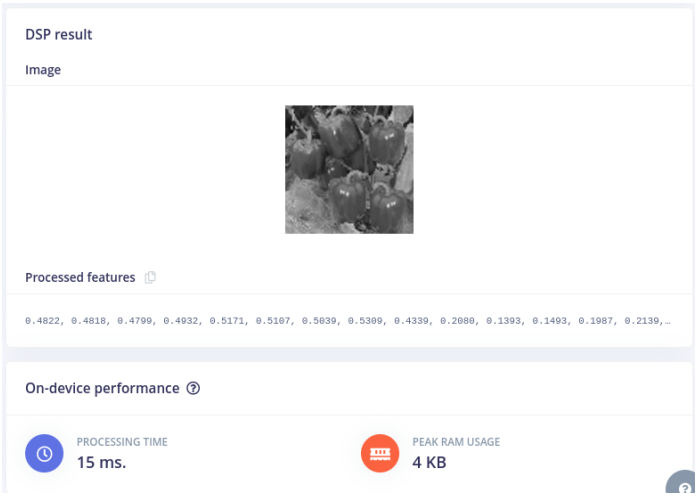
[15] utilizando una arquitectura arquitecturas VGG19 obtuvo un una exactitud del 94.68 %, mientras que [14] utilizando una red neuronal convolucional (ConvNet) de 3 capas convolucionales obtuvo un 96.88 %. Por otro lado, [2] utilizando un conjunto de datos de 2368 imágenes de pimiento morrón y mediante una red neuronal convolucional (ConvNet) clasifica en 3 tipos: pimiento verde, pimiento rojo y pimiento amarillo; logrando un 100 % de exactitud.

No obstante, los enfoques tradicionales emplean modelos neuronales computacionalmente demandantes no optimos para dispositivos de baja potencia. Por lo cual, TinyML utiliza el diseño de modelos de aprendizaje (profundos y no profundos) adecuados para dispositivos de limitado poder computacional. Investigaciones recientes se han centrado en la aplicación de la IA en dispositivos embebidos con el objetivo de implementar modelos de ML en dispositivos con recursos limitados con diversos objetivos [12].

Un ejemplo de ello es [16], donde destaca la importancia de incorporar IA en los dispositivos de IoT mediante la implementación de modelos de ML en hardware de bajo rendimiento. De la misma manera, [16] examina los modelos, estructuras y criterios para integrar el aprendizaje automático de vanguardia en dispositivos de IoT. De esta manera, la IA integrada en dispositivos con recursos limitados ha encontrado numerosas aplicaciones en diversos campos, incluidos la visión por computadora, la atención médica, la robótica, entre otras [1]. Asimismo, revisiones sistemáticas han sintetizado los avances existentes sobre este campo destacando el progreso, los desafíos y trazando una posible ruta futura [8, 19].

En general, la investigación en aplicaciones de IA en dispositivos con recursos limitados abarca varios dominios, desde sensores de IoT energéticamente eficientes hasta la evaluación comparativa de modelos de detección de objetos en dispositivos integrados, mostrando las diversas aplicaciones y avances en el campo. TinyML ha demostrado ser particularmente útil en el seguimiento de hortalizas, monitoreo de sistemas hidropónicos y en general en la agricultura de precisión [23].

Un ejemplo de ello es en [9] donde se presenta una implementación de una red neuronal convolucional, la cual contiene dos capas convolucionales, dos capas de normalización batch y una capa densa, dentro de un dispositivo ESP32-S3 para la inspección post-cosecha del fruto del olivo de la variedad Jordania. Dicha implementación logra una exactitud de 97.39 % utilizando imágenes de 50×50 pixeles.



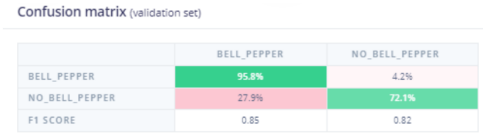
(a) Apartado para configurar los parámetros de las imágenes.

(b) Explorador de características y rendimiento.

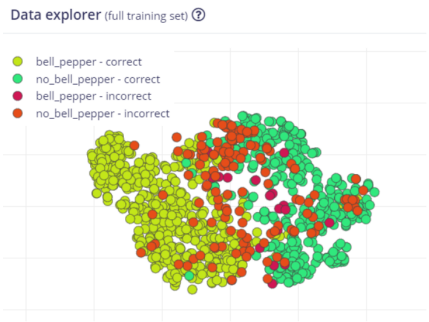
Fig. 5. Bloque de procesamiento de edge impulse.

Por otro lado, [7] donde presenta Lite-Agro, un sistema de identificación de enfermedades en la pera, a través del análisis de imágenes de hojas de pera utilizando TinyML. Para este trabajo se implementó una red neuronal profunda con arquitectura de tipo InceptionV3, inspirada en una arquitectura Xception, el cual logra una exactitud del 99.73 %. Dicho modelo fue entrenado durante 100 epochas utilizando el dataset DiaMOS, el cual contiene 3505 imágenes de hojas de pera divididas en cuatro clases. Finalmente el modelo se implementó en un dispositivo ESP32-CAM.

(a) Precision y valor de pérdida.



(b) Matriz de confusión.



(c) Gráfica de resultados al clasificar las imágenes de entrenamiento.

(d) Resultados de entrenamiento del modelo.

Fig. 6. Resultados del bloque de entranamiento de edge impulse.

En el mismo contexto de TinyML, existen otros trabajos que han empleado redes neuronales para el proceso de inspección y detección de enfermedades en plantas, en su gran mayoría han optado por dispositivos con baja potencia computacional diferentes a microcontroladores como teléfonos móviles y Sistemas en un chip (SoC, del inglés system on a chip) [6, 21, 13]. En la Tabla 1 se presenta un breve resumen del estado del arte de sistemas TinyML aplicadoa a la agricultura.

3. Metodología

El objetivo de este trabajo es detectar la existencia, o no existencia, de pimientos morrón en una imagen mediante una red neuronal convolucional (CNN). La metodología utilizada se divide en cinco etapas: (1) Estudio de la arquitectura de la ESP32CAM, (2) Generación del conjunto de datos, (3) Pruebas de ángulo de visión de la ESP32CAM, (4) Entrenamiento del modelo de Red Neuronal Convolucional (5) Implementación del modelo de aprendizaje en la ESP32CAM. Las etapas realizadas en este trabajo se ilustran en la Figura 1. Para la realización de este trabajo se utilizaron las siguientes herramientas y recursos: el IDE de Arduino, TensorFlow Lite Micro, el sistema Edge Impulse [20] para el diseño y entrenamiento de la red neuronal. Asimismo, la evaluación del sistema se realizó mediante pruebas de rendimiento para determinar la precisión, la sensibilidad y la velocidad de la red neuronal entrenada.

3.1. Arquitectura de la ESP32CAM

La ESP32CAM es un módulo de bajo costo y bajo consumo de energía el cual integra un microcontrolador ESP32 con un sensor de cámara OV2640.

Tabla 2. Hiperparámetros para el entrenamiento del modelo.

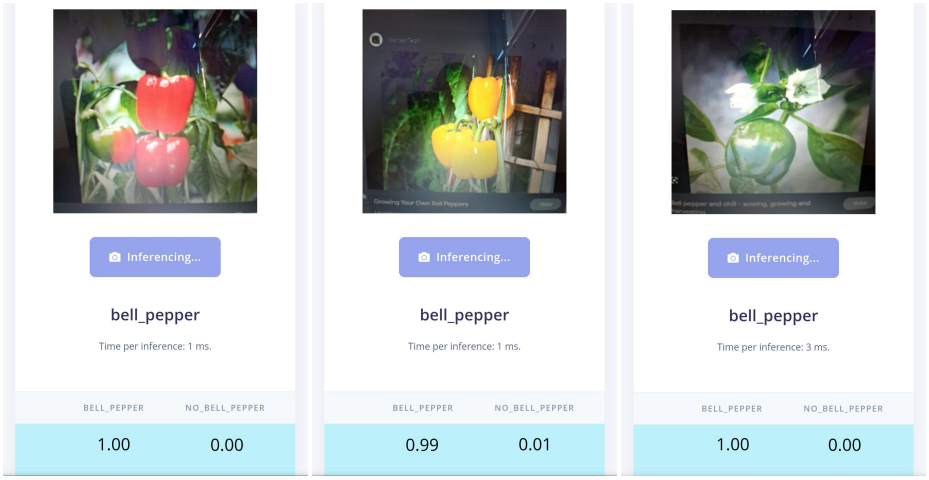
Modelo	lr	MN alpha	Densa	Dropout	exac.	f1	prec.	sens.
MovileNetV2	0.01	0.05	32	0.3	77.55	0.8	0.84	0.77
MovileNetV2	0.001	0.05	32	0.3	81.46	0.84	0.88	0.8
MovileNetV1	0.001	0.25	32	0.1	90.86	0.92	0.95	0.9
MovileNetV1	**0.01**	**0.25**	**16**	**0.1**	**91.12**	**0.93**	**0.95**	**0.91**
MovileNetV1	0.01	0.2	32	0.1	90.08	0.91	0.94	0.89
MovileNetV1	0.001	0.1	32	0.1	83.29	0.87	0.89	0.85
MovileNetV2	0.01	0.35	32	0.1	87.73	0.86	0.93	0.8
MovileNetV2	0.001	0.35	32	0.1	89.56	0.89	0.94	0.84
MovileNetV2	0.001	0.35	32	0.3	87.73	0.89	0.94	0.84

Tiene un microcontrolador de dos núcleos LX6 de 32 bits que operan a una frecuencia de hasta 240 MHz. Tiene una capacidad de 4MB de memoria flash y 520 KB de SRAM. La ESP32CAM cuenta con WiFi, Bluetooth y BLE integrados que permiten la comunicación inalámbrica con otros dispositivos, soporta cámaras OV2640 y OV7670 que puede capturar imágenes de hasta 2 megapixeles [4]. Esta arquitectura la convierte en una plataforma conveniente y de muy bajo costo para la implementación de aplicaciones de visión artificial en el contexto de TinyML.

3.2. Conjunto de datos

Para este trabajo se utilizó un conjunto de datos con las clases "pimiento" y "no-pimiento", dicho conjunto de datos se obtuvo mediante la unión de dos conjuntos de datos públicos de pimiento morrones: el primero se obtuvo de [18], el cual es un conjunto de datos con 3825 imágenes para el reconocimiento de diversas frutas y verduras; el segundo se obtuvo de [10], el cual contiene 1300 imágenes etiquetadas de pimientos morrón. Adicionalmente al conjunto de datos se agregaron imágenes de diversos tipos de árboles, plantas cultivadas en macetas, etc, las cuales se descargaron manualmente de Google. Algunas de las imágenes en el conjunto de datos se muestra en la figura 2. Debido a que las imágenes obtenidas de diversas fuentes tienen distintas dimensiones, éstas se redimensionaron a 96×96 mediante un Script de python.

Posteriormente para el caso de la clase "pimiento", se eliminaron de manera manual imágenes que no correspondiesen íntegramente a pimientos. El conjunto de datos final contiene 1910 imágenes, donde 876 pertencen a la clase "pimiento" y 1034 pertencen a la clase "no-pimiento". Para la etapa de entrenamiento, ver sección 4, el conjunto de datos se dividió en una proporción del 80 % para entrenamiento (1527 imágenes) y el 20 % restante (383 imágenes) para prueba. Sin embargo, y debido a las limitaciones que impone Edge Impulse, no se utilizó una validación cruzada para el caso particular de este trabajo.



(a) Primera prueba. (b) Segunda prueba. (c) Tercera prueba.

Fig. 7. Pruebas en Edge Impulse de la clase “pimiento”.

3.3. Pruebas de ángulo de visión de ESP32CAM

Durante las pruebas iniciales de la ESP32CAM se observó que el ángulo de visión cambia en función de la resolución de ésta, además no cuenta con un sistema de auto-foco por lo que se deberá tener contemplado la resolución de las imágenes a obtener para posicionar el dispositivo a una distancia tal que se logre capturar el fruto en su totalidad y lo mas claro posible. Se realizaron capturas con resoluciones de 96×96, 160×120, 176×144 y 240×176, debido a que el dispositivo ESP32CAM no posee recursos computacionales necesarios para almacenar una red neuronal que clasifique imágenes con mayor resolución. Además, se capturaron a una distancia de 18 y 24 centímetros del objetivo; ésto para realizar una comparación entre las imágenes tomadas a estas dos distancias y poder optar por la más adecuada para una implementación en un sistema real a futuro.

En la figura 3 y en la figura 4 se pueden apreciar las diferencias entre las imágenes capturadas a diferente distancia, también se logra notar cómo el ángulo de visión de las imágenes aumenta al cambiar la resolución de 96×96 a 160×120 píxeles. Sin embargo, al aumentar la resolución a 176×144 aún se nota un ligero aumento en el ángulo de visión pero éste no es tan considerable como el anterior. Con respecto al flash en este caso no se observó ningúna variación debido a las las capturas se realizaron durante el día, no obstante, éste es un aspecto a tener en cuenta para una implementación real a futuro.

4. Entrenamiento de modelo en Edge impulse

En este proyecto se optó por el uso de la plataforma Edge Impulse [20], la cual es una plataforma que facilita el desarrollo e implementación de modelos de aprendizaje automático en dispositivos de borde, como microcontroladores, sensores y FPGAs.

(a) Primera prueba. (b) Segunda prueba. (c) Tercera prueba.

Fig. 8. Pruebas en la ESP32CAM de la clase “pimiento”.

Entre las características que ofrece se encuentran la captura, etiquetado y preparación de datos, selección, entrenamiento y optimización de modelos de aprendizaje. Una vez creado el proyecto en Edge Impulse, y subir el conjunto de imágenes, se procedió a generar un Impuso (Impulse) el cual toma datos crudos, extrae las características y luego usa un bloque de aprendizaje para clasificar nuevos datos. El impulso esta compuesto por 3 secciones principales las cuales se listan a continuación.

– Bloque de entrada (Image data) determina el formato de los datos con los que se está entrenando el modelo.

– Bloque de procesamiento (Processing block) es el bloque donde se realizan la extracción de características y el preprocesamiento de datos.

– Bloque de aprendizaje (Learning block) realiza el proceso de entrenamiento del modelo que se seleccionó, existiendo diversos algoritmos de aprendizaje que incluyen tareas para clasificación, regresión, detección de anomalías, transferencia de imágenes, detección de palabras clave o detección de objetos.

El bloque de procesamiento se configura la transformación de las imágenes en escala de grises, ya que ésto hace que la red neuronal utilice menos recursos a la hora de ser implementada en la ESP32CAM, tal como se muestra en la Figura 5a. Una vez ejecutado el bloque de procesamiento, se muestra el tiempo de ejecución y la cantidad máxima de memoria RAM utilizada además de una comparación del espacio de clasificación de las clases, tal como se muestra en la figura 5b.

En el bloque de aprendizaje se configura la arquitectura de la red neuronal, además de agregar los valores de diversos hiperparámetros para el aprendizaje tales como el número de épocas, la tasa de aprendizaje, el porcentaje del subconjunto de validación, entre otros. Al finalizar el entrenamiento del modelo, Edge Impulse muestra los resultados de precisión, pérdida, matriz de confusión y una gráfica que representa los resultados al clasificar las imágenes del dataset; además presenta el rendimiento del modelo (Figura 6).

(a) Primera prueba. (b) Segunda prueba. (c) Tercera prueba.

Fig. 9. Pruebas en la ESP32CAM de la clase "no-pimiento".

4.1. Implementación de modelo en ESP32CAM

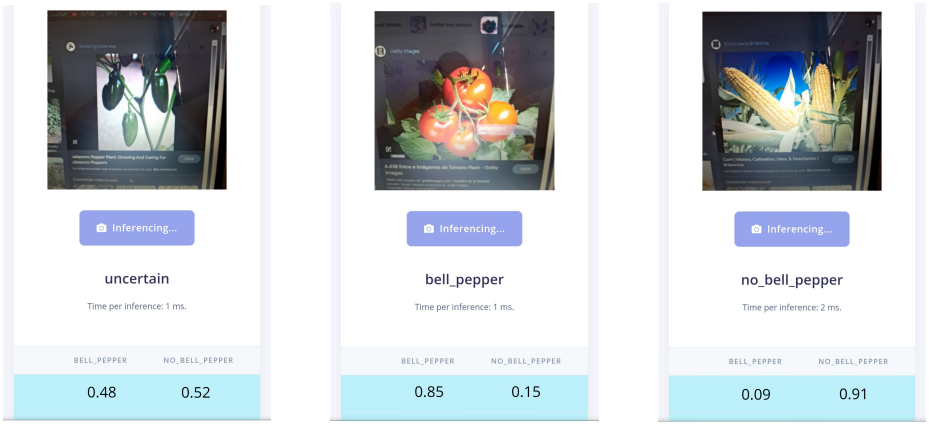
El modelo entrenado se implementó en la ESP32CAM utilizando la plataforma Edge Impulse. En primer lugar, el modelo se exportó como una librería de Arduino, la cual contiene el modelo de la CNN en formato hexadecimal, permitiendo su ejecución en el dispositivo mediante TensorFlow Lite Micro. En segundo lugar se modificó la librería para que la ESP32CAM funcione como un servidor web permitiendo recibir solicitudes HTTP desde un navegador web externo, y poder enviar la imagen capturada de regreso. En resumen, la implementación del modelo en la ESP32CAM se realizó exitosamente, permitiendo la interacción con la cámara y el modelo de clasificación a través de un servidor web.

5. Resultados experimentales

Para este trabajo se realizaron diversos experimentos donde se utilizó la arquitectura de red neuronal convolucional de MobileNet. Se optó por el uso de la arquitectura MobileNet ya que esta diseñada específicamente para ejecutarse en dispositivos con recursos limitados, además de que existen modelos preentrenados con conjuntos de datos como COCO2017, lo que permite disminuir el tiempo de entrenamiento mediante el proceso de entrenamiento de la red. De este modo, cada modelo fue entrenado durante 100 épocas, utilizando un batch size de 32 elementos.

Por cada experimento se varió de manera manual los hiperparámetros de tasa de aprendizaje, versión de la arquitectura MobileNet, el valor del peso alpha de MobileNet, cantidad de neuronas en la capa densa y el porcentaje de dropout. Se estableció una relación $80/20$ para los conjuntos de entrenamiento y prueba. Adicionalmente, se fijó un $20\,\%$ para validación durante el entrenamiento.

Las dos versiones de MobileNet (versión 1, y Versión 2) utilizadas por Edge Impulse están preentrenadas con el conjunto de datos de COCO2017, de esta manera se utiliza la técnica de transferencia de conocimiento (Transfer learning) para fijar los valores de los pesos de las capas convolucionales y sólo optimizar los valores de los pesos en la capa densa.



(a) Primera prueba. (b) Segunda prueba. (c) Tercera prueba.

Fig. 10. Pruebas en Edge Impulse de la clase “no-pimiento”.

Ésto da como consecuencia una disminución en las épocas necesarias y el tiempo de entrenamiento. Finalmente el modelo es quantizado a Int8 para disminuir su tamaño y pueda ser implementado en dispositivos de borde. En la tabla 2 se proporcionan la configuración para el entrenamiento de cada modelo y los resultados obtenidos con el subconjunto de prueba.

5.1. Pruebas mediante Edge impulse

Edge Impulse ofrece una manera sencilla de poner a prueba el modelo a través del navegador web. Inicialmente se probó mediante imágenes de varios objetos que se podrían encontrar en un huerto como herramientas de jardinería, plantas sin ningún fruto, insectos, etc, así como imágenes de pimientos morrones, que son los que se busca identificar. En la figura 7 se muestran algunas pruebas del modelo mediante la página de Edge Impulse para la clase “pimiento”.

Como se puede observar se logra identificar imágenes donde existe presencia de pimientos morrón. En general el modelo aún clasifica de manera incorrecta imágenes de frutos similares a pimientos, hojas de arboles e insectos parados en hojas. En la figura **??**, se puede observar que aúnque el modelo clasifica correctamente muchas imágenes, existen falsos positivos. En la tabla 2 se proporcionan los valores de los hiperparámetros y los resultados obtenidos; se oberva una exactitud de $91{,}12\,\%$ del mejor modelo (ya quantizado a int8) hasta el momento.

5.2. Pruebas en la ESP32CAM

Luego de probar el modelo de clasificación dentro de la plataforma Edge Impulse, se implementó la red neuronal en la ESP32CAM siguiendo el procedimiento descrito en la sección 4.1. Posteriormente, se capturaron múltiples fotografías en el contexto del huerto de pimientos morrón. Algunos de los resultados de las clasificaciones realizadas por el modelo se presentan a continuación.

Estos resultados demuestran la capacidad del sistema para identificar y clasificar objetos en un entorno real, con un alto grado de precisión. En las figuras 8 y 9 se muestran algunos de los resultados de la detección de imágenes capturadas y clasificadas mediante la red neuronal implementada en la ESP32CAM. A pesar de que el sistema detecta la gran mayoría de pimientos morrones, aún se obtienen resultados falsos verdaderos como el que se observa en la figura 9c.

5.3. Discusión de resultados

Al haber realizado las pruebas del modelo tanto en la plataforma de Edge Impulse como con la ESP32CAM, se observó que el modelo entrenado es impreciso, especialmente a la hora de mostrarle otros frutos. No obstante, el sistema logra clasificar imágenes de pimientos morrones de forma correcta. Se sugiere que estos resultados son causa de una baja calidad del conjunto de datos, así como una carente optimización de los hiperparámetros en el entrenamiento del modelo.

De acuerdo a los resultados se pueden observar diversas ventajas: El modelo neuronal se implementó de forma exitosa en el dispositovo ESP32CAM, lo que demuestra la factivilidad del modelo para dispositivos de baja potencia, además dicho modelo muestra una alta exactitud en las muestras de pimientos morrones. Por parte de la ESP32CAM es un dispositivo de bajo costo y de fácil acceso, que conjunto con la plataforma Edge Impulse facilita la implementación de modelos para aplicaciones de TinyML. Por otro lado, se observaron diversas desventajas: En primer lugar, se debe seleccionar y/o implementar un modelo neuronal adecuado para cada dispositivo, ya que en ocasiones el modelo superaba las capacidades de memoría y almacenamiento de la ESP32CAM y no era posible la implementacióno bien, en ocasiones la ESP32CAM se congelaba. De igual manera, en ocasiones el tiempo de respuesta superaba los 5 segundos, que para algunas aplicaciones puede ser un tiempo importante.

Como trabajos futuros se enfocará en mejorar la precisión del modelo, ampliar su funcionalidad y explorar diferentes plataformas de hardware, debido a que la ESP32CAM impone importantes limitaciones como son: a) la admición de redes neuronales convolucionales ligeras como la MobileNet v2 con un máximo de 32 neuronas, b) sólo procesa imágenes en escala de grises con una resolución limitada de 96×96 píxeles. Estas limitaciones hacen todo un reto para la implementación de proyectos en ambientes reales. Tomando en cuenta dichas limitaciones y los resultados obtenidos en este trabajo, se proponen las siguientes líneas de investigación a futuro: I) Expandir la capacidad del modelo para clasificar el nivel de maduración de los pimientos. II) Realizar una comparación del rendimiento de diferentes dispositivos de borde de bajo costo como ESP-Eye, Raspberry Pi Pico y Arduino 33 BLE Sense. III) Evaluar el rendimiento y la precisión de diferentes arquitecturas de redes neuronales "ligeras" como SqueezeNet, MobileNet y ShuffleNet.

6. Conclusiones

En el presente trabajo se realizó un estudio de la arquitectura y el funcionamiento de la ESP32CAM para la detección de pimientos en imágenes mediante una red neuronal MobileNet.

El modelo obtuvo una exactitud de $91{,}12\,\%$ logrando identificar en su mayoría imágenes con pimientos estando presentes, sin embargo, se observan dificultades en la clasificación de objetos como herramientas de jardinería y otros frutos distintos a pimientos morrones. Ésto se le puede atribuir a las limitaciones de la ESP32CAM, como su capacidad limitada para almacenar redes neuronales y procesar imágenes, pero principalmente por la calidad del dataset. Sin embargo, es importante destacar que el modelo no presentó falsos negativos al identificar pimientos morrones, lo que constituye un logro significativo para el desarrollo de un sistema de clasificación de pimientos morrones basado en dispositivos de borde de bajo costo. Los resultados mostrados en este trabajo alentadores para la implementación de un sistema de monitoreo autónomo y de bajo costo para sistemas agricolas urbanos y/o sin suelo.

Referencias

1. Ajani, T. S., Imoize, A. L., Atayero, A. A.: An overview of machine learning within embedded and mobile devices–optimizations and applications. Sensors, vol. 21, no. 13, pp. 4412 (2021) doi: 10.3390/s21134412
2. Almadhoun, H. R.: Bell pepper classification using deep learning. International Journal of Academic Engineering Research, vol. 5, no. 1, pp. 75–79 (2021)
3. Bassie, H., Sirany, T., Alemu, B.: Rural-urban labor migration, remittances, and its effect on migrant-sending farm households: Northwest Ethiopia. Advances in Agriculture, vol. 2022, pp. 1–8 (2022) doi: 10.1155/2022/4035981
4. DFROBOT: Esp32-cam development board (2020) www.dfrobot.com/product-1879.html
5. Dhawi, F.: The role of plant growth-promoting microorganisms (PGPMs) and their feasibility in hydroponics and vertical farming. Metabolites, vol. 13, no. 2, pp. 247 (2023) doi: 10.3390/metabo13020247
6. Dhiman, P., Kaur, A., Hamid, Y., Alabdulkreem, E., Elmannai, H., Ababneh, N.: Smart disease detection system for citrus fruits using deep learning with edge computing. Sustainability, vol. 15, no. 5, pp. 4576 (2023) doi: 10.3390/su15054576
7. Dockendorf, C., Mitra, A., Mohanty, S. P., Kougianos, E.: Lite-agro: Exploring light-duty computing platforms for IoAT-edge AI in plant disease identification. Internet of Things Advances in Information and Communication Technology, pp. 371–380 (2023) doi: 10.1007/978-3-031-45882-8_25
8. Han, H., Siebert, J.: TinyML: A systematic review and synthesis of existing research. In: International Conference on Artificial Intelligence in Information and Communication, pp. 269–274 (2022) doi: 10.1109/ICAIIC54071.2022.9722636
9. Hayajneh, A. M., Batayneh, S., Alzoubi, E., Alwedyan, M.: TinyML olive fruit variety classification by means of convolutional neural networks on IoT edge devices. AgriEngineering, vol. 5, no. 4, pp. 2266–2283 (2023) doi: 10.3390/agriengineering5040139
10. images.cv: 1.3k bell pepper labeled image dataset (2020) images.cv
11. Instituto Nacional de Estadística y Geografía: Población rural y urbana (2020) cuentame.inegi.org.mx/poblacion/rur_urb.aspx?tema=P
12. Kwon, J., Park, D.: Hardware/software co-design for TinyML voice-recognition application on resource frugal edge devices. Applied Sciences, vol. 11, no. 22, pp. 11073 (2021) doi: 10.3390/app112211073
13. Liu, C., Wang, X., Ni, J., Cao, Y., Liu, B.: An edge computing visual system for vegetable categorization. In: Proceedings of the 18th IEEE International Conference On Machine Learning And Applications, pp. 625–632 (2019) doi: 10.1109/ICMLA.2019.00115

14. Mahamud, F., Neloy, M. A. I., Barua, P., Das, M., Nahar, N., Hossain, M. S., Andersson, K., Hoassain, M. S.: Bell pepper leaf disease classification using convolutional neural network. In: International Conference on Intelligent Computing and Optimization, vol. 569, pp. 75–86 (2022) doi: 10.1007/978-3-031-19958-5_8
15. Mathew, M. P., Elayidom, S., Jagathyraj, V.: Disease classification in bell pepper plants based on deep learning network architecture. In: Proceedings of the 2nd International Conference for Innovation in Technology, pp. 1–6 (2023) doi: 10.1109/INOCON57975.2023.10101269
16. Merenda, M., Porcaro, C., Iero, D.: Edge machine learning for AI-enabled IoT devices: A review. Sensors, vol. 20, no. 9, pp. 2533 (2020) doi: 10.3390/s20092533
17. Raza, W., Osman, A., Ferrini, F., Natale, F. D.: Energy-efficient inference on the edge exploiting TinyML capabilities for UAVs. Drones, vol. 5, no. 4, pp. 127 (2021) doi: 10.3390/drones5040127
18. Seth, K.: Fruits and vegetables image recognition dataset (2020) www.kaggle.com/datasets/kritikseth/fruit-and-vegetable-image-recognition
19. Shafique, M., Theocharides, T., Reddy, V. J., Murmann, B.: TinyML: Current progress, research challenges, and future roadmap. In: Proceedings of the 58th ACM/IEEE Design Automation Conference, pp. 1303–1306 (2021) doi: 10.1109/DAC18074.2021.9586232
20. Shelby, Z., Jongboom, J., Huang, S., Watkins, L., DeLey, W.: Edge impulse (2024) www.edgeimpulse.com
21. Silva, M., Ferreira-da-Silva, J., Oliveira, R.: IDiSSC: Edge-computing-based intelligent diagnosis support system for citrus inspection. In: Proceedings of the 23rd International Conference on Enterprise Information Systems, pp. 685–692 (2021) doi: 10.5220/0010444106850692
22. Susanto, F., Suryani, N. K., Darmawan, P., Prasiani, K., Ramayu, I. M. S.: Comprehensive review on automation in hydroponic agriculture using machine learning and IoT. RSF Conference Series: Engineering and Technology, vol. 1, no. 2, pp. 86–95 (2021) doi: 10.31098/cset.v1i2.479
23. Tang, Y., Chen, M., Wang, C., Luo, L., Li, J., Lian, G., Zou, X.: Recognition and localization methods for vision-based fruit picking robots: A review. Frontiers in Plant Science, vol. 11 (2020) doi: 10.3389/fpls.2020.00510
24. Tunio, M. H., Gao, J., Mohamed-Tarek, M. K., Ahmad, F., Abbas, I., Ali-Shaikh, S.: Comparison of nutrient use efficiency, antioxidant assay, and nutritional quality of butter-head lettuce (Lactuca sativa L.) in five cultivation systems. International Journal of Agricultural and Biological Engineering, vol. 16, no. 1, pp. 95–103 (2023) doi: 10.25165/j.ijabe.20231601.6794

ISSN 1870-4069

Aplicación Android para clasificar señalamientos en campus universitario usando aprendizaje de máquina

Elohim Ramírez-Galván, Cesar Benavides-Alvarez,
Carlos Avilés-Cruz, Arturo Zúñiga-López

Universidad Autónoma Metropolitana,
Unidad Azcapotzalco,
México

{al2223800666, cesarbenavides, caviles,
azl}@azc.uam.mx

Resumen. El poder identificar correctamente la señalización en un campus universitario es de suma importancia para estar informado y poderse ubicar y desplazar dentro del mismo. En el presente trabajo se plantea el desarrollo, implementación y pruebas de un sistema clasificador de imágenes en una aplicación para dispositivos móviles Android. El sistema obtiene un vector tridimensional a partir de los canales RGB (Red Green Blue) de la imagen para poder clasificarla mediante la evaluación del vector con clasificadores distintos (Coeficiente de correlación, Bayes, Perceptrón, K-próximos vecinos (KNN)), mismos que devuelven una selección de clase y que el sistema elige la mejor representada. El sistema propuesto tiene una eficiencia de 99.67 % en la correcta clasificación de la señalización.

Palabras clave: Clasificador binario, perceptrón, bayes, kNN, clasificador de imágenes.

Android Application to Classify Signage on University Campuses Using Machine Learning

Abstract. Being able to correctly identify the signage on a university campus is of utmost importance in order to be informed and to be able to locate and move around within it. In this work, the development, implementation, and testing of an image classifier system in an application for Android mobile devices are proposed. The system obtains a three-dimensional vector from the RGB image channels in order to classify it by evaluating the vector with different methods (Correlation Coefficient, Bayes, Perceptron, K-nearest neighbors (KNN)), which return a class selection, and the system chooses the best-represented one. The proposed system has a 99.67% efficiency in the correct classification of the signaling.

Keywords: Binary classifier, perceptron, bayes, kNN, image classifier.

1. Introducción

Dentro del ámbito de la Inteligencia Artificial, y en particular, en el aprendizaje de máquina, el reconocimiento de imágenes es una rama de crecimiento con diversas aplicaciones en múltiples campos. Existen diversas maneras en las que esto se muestra, dependiendo del campo y objetivo. El presente trabajo consiste en el desarrollo, implementación y pruebas de un clasificador de señalizaciones pertenecientes a la Universidad Autónoma Metropolitana Unidad Azcapotzalco (UAM-A), cuyo objetivo es distinguir entre dos tipos de señales (clases) que forman parte de un grafo (como nodos) para la navegación dentro del campus universitario.

De esta manera la navegación móvil proporciona una manera conveniente y eficaz de encontrar direcciones y acceder a una variedad de servicios relacionados con la ubicación, convirtiéndola en una herramienta indispensable en la vida diaria. En este trabajo se presenta un sistema clasificador binario que distingue entre señales de tipo Directorio (Dir) y señales de tipo Punto de Reunión (PR), mostradas en la Figura 1. Cada una de las clases cuenta con características fácilmente distinguibles entre las cuales destaca el color.

El sistema propuesto aprovecha esta característica para diferenciar ambas clases de manera efectiva. El clasificador binario está disponible para dispositivos móviles con sistema operativo Android debido a que, según el Instituto Federal de Telecomunicaciones (IFT) en la Cuarta Encuesta 2020, de Usuarios de Servicios de Telecomunicaciones, más del 80 % de usuarios de telefonía móvil usan este Sistema Operativo, por lo que se pretende que el sistema pueda ser usado por un mayor número de usuarios. El resto del artículo está constituido del estado del arte capítulo 2, la Metodología, descrita en el capítulo 3. En el capítulo 4 se describe la experimentación y resultados; finalmente, las conclusiones y perspectivas son descritas en el capítulo 5.

2. Estado del arte

Existe numerosa bibliografía relacionada al reconocimiento de imágenes, siendo en su mayoría realizada con redes neuronales convolucionales (CNN), que son bien conocidas por los buenos resultados que ofrece, sin embargo, para ello se suele requerir de un amplio conjunto de imágenes, además de numeroso tiempo de entrenamiento y recursos en el equipo de cómputo que utiliza, así como en los amplios conocimientos en el tema para su correcto desarrollo. Es por ello, que la revisión del trabajo relacionado se centra en clasificadores (que no sean CNN) cuyos datos de entrada son imágenes y entre cuyas características principales para la clasificación se encuentre el color.

En [1] se realizó un sistema que permite la detección de enfermedades torácicas a partir de imágenes de Rayos X. Las imágenes pasan por un preprocesamiento para luego discernir entre paciente sano o enfermo. Cuando el resultado es “enfermo”, pasa por 16 clasificadores más, cada uno asociado a una enfermedad en particular, y de la cual se obtiene una probabilidad de que la imagen corresponda a la enfermedad en sí. Al final, el sistema muestra la enfermedad de la probabilidad más alta para la imagen ingresada. Todos los clasificadores están basados en DenseNet. En [4], se presentó un sistema para reconocer señales de tránsito en Tailandia.

Fig. 1. Ejemplo de señalizaciones de PR (izquierdo) y Dir (derecho).

El sistema consta de dos procesos principales, el primero es un clasificador que determina el tipo de señal a la que corresponde de entre 4 posibles, para ello hace uso de las características obtenidas por medio de extracción usando histograma de gradientes orientados (HOG) y descriptor de capas de color (CLD), mediante una máquina de soporte vectorial (SVM) y random forest “bosques aleatorios” (RF) para este propósito. Posteriormente, sigue una etapa de reconocimiento de signo que se encuentra dentro de dicha señal.

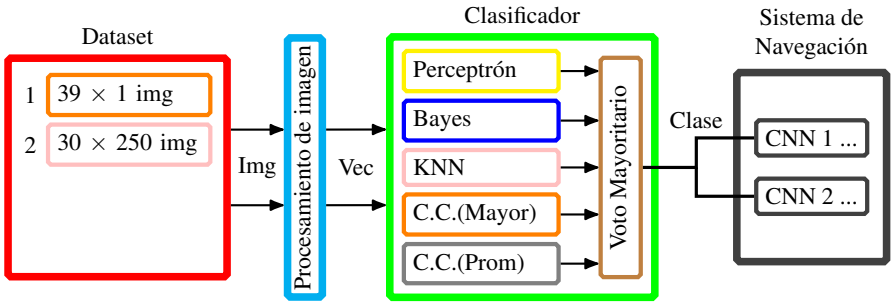


Fig. 2. Modelo de clasificación binario.

En [7], se propuso un método de visión por computadora (usando MATLAB) que permite reconocer frutas, y la idea es que al combinarse con una báscula se pueda obtener el costo respectivo. Esto se realiza a partir de la detección de su forma, color y tamaño. Primero, se realiza un preprocesamiento de la imagen, y una vez extraídas las características de interés, se pasan por diversos clasificadores (KNN, RF, Análisis de datos (DA), SVM), en donde se encontró que KNN tuvo una mejor precisión.

En [5], se presentó un sistema de Visión por computadora, que usando Matlab, distingue entre dos clases de pistaches. Los autores crearon su propio conjunto de datos con una cámara de alta resolución y de las cuales se extrajeron 16 características de interés, mismas que se usaron para la clasificación de los pistaches usando KNN.

En [6], se desarrolló un algoritmo de análisis de imagen para la clasificación de cerezas. Primero se realiza una segmentación de la imagen para dejar únicamente a la fruta, luego se obtiene el histograma de la imagen en formato RGB y Hue Saturation Value (HSV), donde R y H son los componentes que presentan diferencias para la clasificación, y se usan en un clasificador bayesiano para determinar la pertenencia a una de las 5 clases existentes, obteniendo una precisión del 100 %. La implementación se realizó en una computadora con velocidad de procesador de 300 MHz.

En [9], se presentó un clasificador que usa KNN para poder delimitar a la fruta del dragón en la imagen, pues existen diferentes colores para los distintos tipos en diferentes áreas y condiciones de crecimiento, mientras que usa CNN para extraer las características externas de la fruta (como dimensiones y defectos), y finalmente clasificarla entre los 3 grupos posibles.

En [8], se presentó un sistema que detecta las manchas en las hojas de algunas plantas mediante extracción de características como el color y la textura comparando dos clasificadores: SVM optimizado con Bayes y RF. En [3], se propuso un proceso de clasificación que permite determinar el nivel de madurez de la naranja basado en las características de su cáscara (como el color), para ello se utiliza HSV, así como otros métodos para la textura, y siendo clasificados mediante KNN.

En [2], se determinó el nivel de enfermedad de plantas en función de las manchas en sus hojas usando extracción de características como la textura usando Gray-Level Run-Length Matrix (GLRLM) y Gray-Level Ocurrence MAtrix (GLCM), y color (RGB, HSV, Lab) para desarrollar modelos de clasificación usando SVM, KNN y CNN, y donde se encontró que los mejores resultados fueron para la combinación GLRLM-HSV, clasificados con KNN para obtener los mejores resultados.

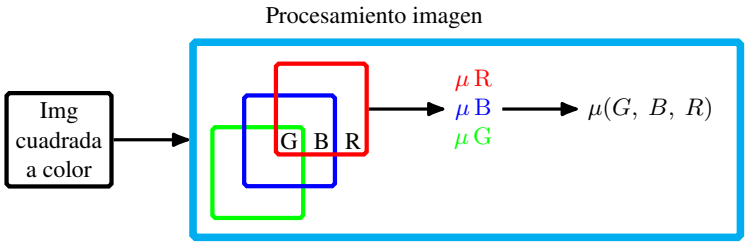


Fig. 3. Procesamiento de imagen de entrada.

3. Metodología

Como ya se mencionó anteriormente, el clasificador binario que se presenta, forma parte de un sistema más grande usado para navegación dentro de la UAM-A. En la Figura 2 se encuentra el esquema del clasificador binario, el cual consiste básicamente en dos etapas, donde la primera es el procesamiento de la imagen, y la segunda corresponde al proceso de clasificación, además del correspondiente conjunto de datos (dataset) necesario para el entrenamiento y evaluación.

3.1. Conjunto de datos

La UAM-A cuenta con un total de 39 señalizaciones de interés (14 PR y 25 Dir) distribuidas en todo el campus, fotografiadas cuidando el enfoque y encuadre adecuado, con un teléfono Xiaomi Poco X3 Pro usando la cámara nativa con tamaño 3000×3000 pixeles, en formato jpg. Este conjunto inicial, fue utilizado para el entrenamiento de los modelos descrito más adelante. Debido a la similitud entre algunas señalizaciones (por ejemplo, algunos edificios cuentan con 2 Dir), estas fueron agrupadas como una misma clase con el fin de simplificar el sistema de navegación, del cual forma parte el clasificador presentado en este trabajo.

De igual forma, algunas otras fueron descartadas debido a la relevancia que tenían, con el mismo propósito. Finalmente quedaron 30 señales diferentes (14 PR y 16 Dir) mostradas en la Figura 1, y que sirven como nodos para el sistema de navegación, y de las cuales se tomaron más de 250 fotografías de cada una, con variaciones de cercanía, iluminación, y perspectiva. Este segundo conjunto de datos, se usó (seleccionando de manera aleatoria entre las imágenes donde la señal no estuviera alejada) para realizar las pruebas una vez teniendo los modelos entrenados.

3.2. Procesamiento de imagen

Las entradas, tanto para el sistema como para el entrenamiento consiste en una imagen cuadrada a color de 1000×1000 pixeles (aunque puede trabajar con diferentes resoluciones), en formato png, la cual es descompuesta en sus 3 canales RGB, y obteniendo el valor promedio de cada canal. Por lo que al final de este proceso se obtiene un vector tridimensional por cada imagen. Este proceso se representa en la Figura 3.

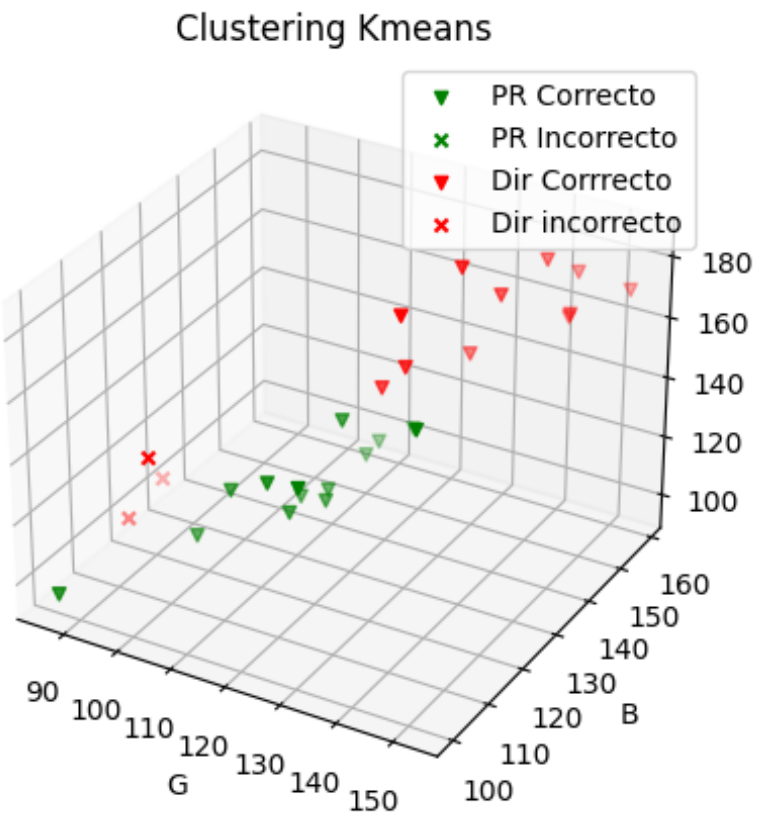


Fig. 4. Agrupamiento usando kmeans.

3.3. Clasificadores

Todos los clasificadores requieren de un entrenamiento a fin de ajustarse a las entradas y resultados deseados. Para realizarlo es necesario un conjunto de datos inicial, que en este caso corresponde a 39 imágenes (una por cada señalización reconocible del modelo de navegación). De estas últimas se descartaron 11 de manera aleatoria para que quedaran ambas clases del mismo tamaño. De modo que al procesar estas imágenes se obtienen 14 vectores para una de las dos clases. Esta misma base sirve para el entrenamiento de los clasificadores, como se indica a continuación:

Perceptrón. Debido a que únicamente son dos clases, se trata de un problema linealmente separable y por tanto, con un único perceptrón. Además, al tratarse de vectores tridimensionales, el perceptrón devolverá la ecuación del plano separatriz entre ambas clases, como se muestra en la Ecuación 1:

$$y = x_1 \times w_1 + x_2 \times w_2 + x_3 \times w_3 + \theta, \tag{1}$$

donde w corresponde a los pesos para cada eje, en este caso, los canales RGB, y x los mismos canales del vector a evaluar.

Fig. 5. Interfaz de la aplicación android.

Y que al evaluar, que el resultado sea mayor o igual que cero, o menor, determinará la clase de pertenencia. El entrenamiento se inició con todos los parámetros en 1 y la base fue normalizada para el entrenamiento, llegando a la ecuación del plano en la iteración 2856. Resultado del entrenamiento: Ecuación del plano separatriz:

$$y = -2{,}29\ B - 4{,}06\ G + 9{,}07\ R - 2. \tag{2}$$

Bayes. La Ecuación 3 para obtener la probabilidad de un vector x en $\mathbb{R}^n$ pertenencia a una clase es la siguiente:

$$P(x/C_k) = (2\pi)^{-n/2}|\Sigma_k|^{-1/2}e^{-1/2(x-\mu_k)\Sigma_k^{-1}(x-\mu_k)^T}. \tag{3}$$

Sin embargo cuando es una clasificación binaria se puede simplificar como se muestra en la ecuación 4:

$$\ln\frac{|\Sigma_1|}{|\Sigma_2|} + (x-\mu_1)\Sigma_1^{-1}(x-\mu_1)^T - (x-\mu_2)\Sigma_2^{-1}(x-\mu_2)^T > 0 \Rightarrow x \in C_1. \tag{4}$$

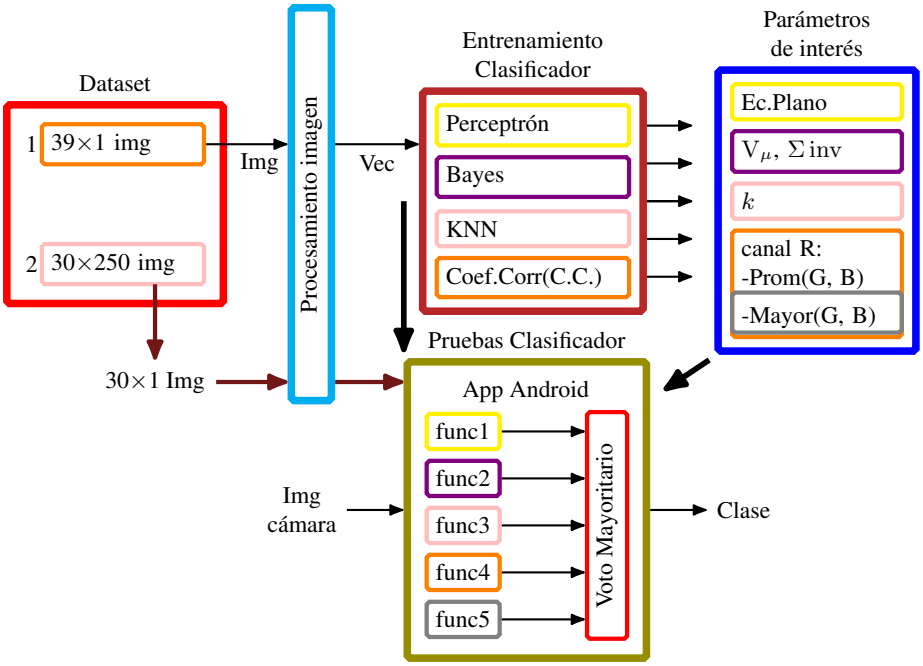


Fig. 6. Interfaz de la aplicación android.

Resultado del entrenamiento: Vectores promedio por clase:

- $\mu_1 = (114{,}2,\ 125{,}0,\ 120{,}9)$.
- $\mu_2 = (137{,}1,\ 142{,}8,\ 168{,}2)$.

Logaritmo del cociente de determinantes:

$$\ln \frac{|\Sigma_1|}{|\Sigma_2|}. \tag{5}$$

Matriz inversa por clase:

$$\Sigma_1^{-1}, \tag{6}$$

$$\Sigma_2^{-1}. \tag{7}$$

KNN. Para este clasificador se realizó una prueba con $k = 5$, bajo el criterio de k impar $<=$ a la raíz cuadrada del número de muestras, probándose los KNN de cada subconjunto de atributos posible (G, B, R, GB, BR, GR, GBR), en donde se obtuvieron los mejores resultados (100 %) en todos aquellos donde aparece el canal R. Se probó también, calcular la distancia del vector completo (RGB), así como únicamente del canal rojo (R) sin que hubiera diferencias entre ambos. Esto puede apreciarse en las figuras 7 y 5, que aparecen en el clasificador como "KNN" y "-KNNRed" respectivamente. Resultado del entrenamiento: Base de valores promedio:

$$k = 5. \tag{8}$$

Tabla 1. Matriz de confusión por modelo.

		Perceptrón		**Bayes**		**Mayor**		**Promedio**		**KNN**		**Final**	
Total	**Tipo**	**PR**	**Dir**	**PR**	**Dir**	**PR**	**Dir**	**PR**	**Dir**	**PR**	**Dir**	**PR**	**Dir**
14	PR	14	0	14	0	14	0	14	0	14	0	14	0
16	Dir	1	15	1	15	1	15	4	12	1	15	1	15

Coeficiente de Correlación. Se calculó el Coeficiente de correlación entre los atributos y la clase usando la Ecuación 9:

$$R_{ij} = \frac{C_{ij}}{\sqrt{C_{ii}\, C_{jj}}}, \tag{9}$$

donde se obtuvo que el canal R, es el que está fuertemente relacionado con la clase de pertenencia (como también se puede apreciar en KNN y en la ecuación del perceptrón), por lo que una vez teniendo esta información se revisaron los vectores de la base a fin de encontrar alguna expresión matemática que pudiera realizar la clasificación en función de sus canales, de las cuales resultaron dos: Resultado del entrenamiento: Pertenece a Dir si:

$$R \text{ es } 10\,\% > \text{prom}(G,\ B), \tag{10}$$

$$R \text{ es } 5\,\% > \text{máx}(G,\ B). \tag{11}$$

K-means. Adicionalmente a los clasificadores anteriores, se realizó clustering usando kmeans para ver si las clases lograban diferenciarse por sí mismas, logrando una precisión mayor al 89 %. En la Figura 4 se muestra la gráfica del agrupamiento resultante, con únicamente 3 errores al clasificar Dir, y el 100 % para PR.

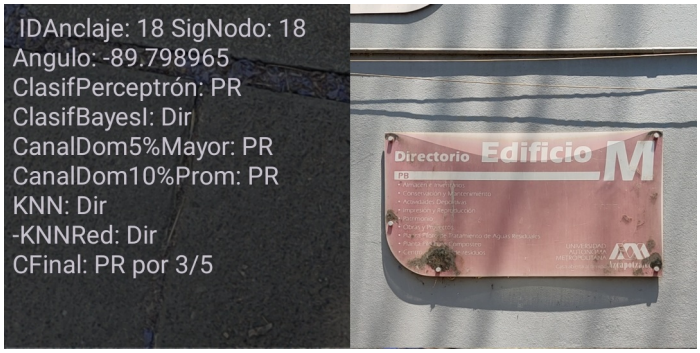
Voto mayoritario. Finalmente, para tomar la decisión de cuál es la clase a la que pertenece la señal registrada, se ingresa el vector obtenido del procesamiento a todos los clasificadores, que a su vez devuelven una clase. Se toma como clase final de pertenencia aquella que tenga al menos 5 coincidencias.

3.4. Aplicación.

El clasificador está implementado en una aplicación Android realizada en Android Studio Giraffe — 2022.3.1 Patch 2. La aplicación cuenta con una pantalla única en la que se visualiza la cámara (previa aceptación de permisos). Esta sección está delimitada con un cuadrado a fin de que la imagen capturada corresponda con las imágenes de entrenamiento, y de este modo el usuario encuadre la señal en este espacio, restringiendo el espacio de pixeles no pertenecientes al señalamiento.

En la parte inferior se ubica un botón con la función de tomar la fotografía, convertir la imagen al vector de colores correspondiente, y pasarlo por las 5 funciones clasificadoras, cuyos resultados individuales y mayoritario se despliegan en un campo de texto, así como la imagen capturada en el campo de imagen correspondiente. En la figura 5 se muestra la interfaz de la aplicación.

(a) Clasificación correcta PR.



(b) Clasificación incorrecta Dir.

Fig. 7. Clasificación de señalización desde aplicación android.

Cada clasificador se realizó a modo de una función independiente que toma como entrada un vector tridimensional y devuelve un valor booleano asociado a la clase de pertenencia. En la Figura 6 se muestra el proceso seguido para la implementación de las funciones (una por clasificador), para todas ellas, se requiere un vector tridimensional como valor de entrada, devolviendo un valor binario correspondiente a la clasificación. Esos valores son los que paran a la sección de Voto mayoritario para su evaluación.

4. Resultados

Una vez obtenidos los resultados del entrenamiento, se realizaron pruebas con imágenes distintas a las del conjunto inicial, obteniéndose una precisión del 96.67 %. Para esta se seleccionó aleatoriamente una imagen por clase, de entre las 30 existentes para el sistema de navegación. En la Tabla 1 se detalla la matriz de confusión. Finalmente se implementó el clasificador en una aplicación Android que permite tomar una fotografía y despliega en pantalla el resultado de todos los clasificadores, incluyendo el voto mayoritario. La figura 7 muestra al clasificador operando en la aplicación Android, y se puede apreciar un ejemplo de clasificación correcta, así como una incorrecta.

Con esta prueba, se logró identificar que las señalíticas erróneamente clasificadas corresponden a las del tipo Dir, que se encuentran a la intemperie, y que presentan por tanto desgaste en comparación a las otras. Este desgaste se aprecia principalmente a modo de decoloración que atenúa los tonos rojos de la misma, y que resultan clave para la correcta clasificación. Sin embargo, este problema sólo resulta cuando el encuadre de la señalización no es adecuado.

Si se compara la clasificación errónea de la figura 7, con el ejemplo existente en la figura 5, se puede apreciar que se trata de la misma señalización, pero que en esta última figura, el encuadre se realizó mejor que en la primera, por lo que la clasificación en ese caso es adecuada. Y de este modo, se alcanza la precisión del 100 % en la clasificación. Este hallazgo es importante porque permite establecer el funcionamiento en entornos donde la variación en el desgaste de las señales sea mínima o controlada, e incluso en casos como la UAM-A donde la mayor parte de las señalíticas se encuentran en buen estado, ofrece una precisión en la clasificación bastante aceptable, aún cuando la toma de las imágenes no sea la ideal.

5. Conclusiones y trabajo futuro

Este trabajo presenta el desarrollo, implementación y pruebas de un sistema clasificador binario de imágenes correspondiente a los tipos de señalización (PR y Dir) de la UAM-A, necesario como módulo para un sistema de navegación. Al ser un sistema diseñado para dispositivos móviles, este debe ser ligero, rápido y preciso, por lo que debe ser lo más simple posible. Es por ello que si bien el sistema trabaja con una imagen como entrada, se realiza un procesamiento con la misma hasta obtener un vector tridimensional de ella, mismo que ingresa a diferentes clasificadores ya reducidos a expresiones algebraicas sencillas para obtener un resultado, mismo que será comparado con el de los demás clasificadores a fin de devolver aquel con una mejor representación del mismo.

El sistema de clasificación resultó altamente efectivo, con precisión superior al 99.6 %, únicamente fallando en señalizaciones visiblemente desgastadas, pero en las que, de tomar las imágenes con un encuadre adecuado se alcanza la precisión del 100 %. Para un trabajo futuro se aumentará otro tipo de clasificadores basados en HSV en lugar de RGB pero con el mismo principio presentado, a fin de ver si se encuentran mejoras que puedan disminuir los errores encontrados. También se pretende eliminar el botón para tomar la fotografía, de modo que la predicción se pueda hacer de manera periódica cada cierto intervalo, a fin de poder mejorar las aplicaciones basadas en el clasificador.

Referencias

1. Agarwal, A., L, V., Paduri, A. R., Mabiyan, R., Wattamwar, M. S., L, D., Darapaneni, N.: Detection of thoracic diseases using chest X-ray: A comparative study of binary class and multiclass classification using deep learning. In: Proceedings of the IEEE Pune Section International Conference, pp. 1–6 (2023) doi: 10.1109/punecon58714.2023.10450111
2. Hayit, T., Endes, A., Hayit, F.: KNN-based approach for the classification of fusarium wilt disease in chickpea based on color and texture features. European Journal of Plant Pathology, vol. 168, no. 4, pp. 665–681 (2023) doi: 10.1007/s10658-023-02791-z

3. Mentari, M., Rahmad, C., Muchlisin, M. S., Sukmana, S. E.: Classification of siam orange ripeness level using k-nearest neighbors algorithm and features gray level run length matrix. In: IEEE International Conference on Communication, Networks and Satellite, pp. 272–277 (2023) doi: 10.1109/comnetsat59769.2023.10420620
4. Namyang, N., Phimoltares, S.: Thai traffic sign classification and recognition system based on histogram of gradients, color layout descriptor, and normalized correlation coefficient. In: Proceedings of the 5th International Conference on Information Technology, vol. 14, pp. 270–275 (2020) doi: 10.1109/incit50588.2020.9310778
5. Ozkan, I. A., Koklu, M., Saraçoğlu, R.: Classification of pistachio species using improved K-NN classifier. Progress in Nutrition, vol. 23, no. 2, pp. e2021044 (2021) doi: 10.23751/pn.v23i2.9686
6. Reyes, J. F., Contreras, E., Correa, C., Melin, P.: Image analysis of real-time classification of cherry fruit from colour features. Journal of Agricultural Engineering, vol. 52, no. 4 (2021) doi: 10.4081/jae.2021.1160
7. Shakya, S.: Analysis of artificial intelligence based image classification techniques. Journal of Innovative Image Processing, vol. 2, no. 1, pp. 44–54 (2020) doi: 10.36548/jiip.2020.1.005
8. Singh, A. K., Sreenivasu, S., Mahalaxmi, U. K., Sharma, H., Patil, D. D., Asenso, E.: Hybrid feature-based disease detection in plant leaf using convolutional neural network, Bayesian optimized SVM, and random forest classifier. Journal of Food Quality, vol. 2022, pp. 1–16 (2022) doi: 10.1155/2022/2845320
9. Trieu, N. M., Thinh, N. T.: A study of combining KNN and ANN for classifying dragon fruits automatically. Journal of Image and Graphics, vol. 10, no. 1, pp. 28–35 (2022) doi: 10.18178/joig.10.1.28-35

ISSN 1870-4069

Evaluación causal de características en base a explicaciones de clasificadores profundos de imágenes médicas: Un estudio de caso sobre imágenes de cálculos renales ex-vivo

Armando Villegas-Jimenez[1], Daniel Flores-Araiza[2], Francisco Lopez-Tiro[2, 3], Miguel Gonzalez-Mendoza[2], Gilberto Ochoa-Ruiz[2], Christian Daul[3]

[1] Instituto Politécnico Nacional,
México

[2] Tecnologico de Monterrey,
México

[3] Université de Lorraine,
Centre National de la Recherche Scientifique,
Francia

Resumen. Entender las razones detrás de las salidas de los modelos de aprendizaje profundo es crucial en el diagnóstico médico. Aunque existen métodos de inteligencia artificial explicable (XAI) para identificar las causas detrás de las predicciones, las evaluaciones cuantitativas de estas relaciones causales son limitadas. Por ello, proponemos una técnica para medir la relación causal entre las características del área de interés en imágenes de una clase específica y la salida de un clasificador, enfocándonos en imágenes de piedras renales. Nuestro método, llamado Puntuación de Explicación Causal (PEC), se evaluó en un conjunto de datos de imágenes ex-vivo de cálculos renales. Los experimentos demostraron que las relaciones causales medidas son más precisas cuando el área de interés se identifica utilizando un método explicable en lugar de anotaciones humanas en cuadros delimitadores, lo que ayuda a identificar qué explicaciones de los resultados de los modelos de aprendizaje profundo son más confiables en el contexto médico. El método PEC adapta técnicas existentes para trabajar con máscaras de segmentación en lugar de cajas delimitadoras, permitiendo una medición más precisa de las relaciones causales. Además, hemos modificado el método GradCAM para automatizar la extracción de máscaras de segmentación binarias, facilitando la obtención de medidas causales más consistentes que con segmentaciones manuales y facilitando el uso de nuetsro metodo al reducir la dependencia de anotaciones humanas. Los resultados indican que el método PEC permite una evaluación más informada de si las predicciones de un modelo y sus explicaciones se derivan de relaciones causales discernibles o no, lo que indica una dirección prometedora para mejorar el nivel de comprensión y confianza que podemos obtener al usar modelos DL como herramientas parael Diagnóstico Asistido por Computadora (CADx).

Palabras clave: Diagnóstico asistido por computadora (CADx), IA explicable (XAI), aprendizaje profundo (DL), análisis de imágenes médicas, análisis morfoconstitucional (MCA), piedras renales.

Armando Villegas-Jiménez, Daniel Flores-Araiza, Francisco López-Tiro, et al.

Causal Evaluation of Features from Explanations of Deep Classifiers of Medical Images: A Case Study on Ex-vivo Kidney Stone Imaging

Abstract. Understanding the reasons behind deep learning models' outputs is crucial in medical diagnosis. Although explainable artificial intelligence (XAI) methods exist to identify the causes behind predictions, quantitative evaluations of these causal relationships are limited. Therefore, we propose a technique to measure the causal relationship between the characteristics of the area of interest in images of a specific class and the output of a classifier, focusing on images of kidney stones. Our method, called Causal Explanation Scoring (PEC), was evaluated on a data set of ex-vivo images of kidney stones. Experiments demonstrated that measured causal relationships are more accurate when the area of interest is identified using an explainable method rather than human annotations of bounding boxes, helping to identify which explanations of the outputs of deep learning models are more reliable in the medical context. The PEC method adapts existing techniques to work with segmentation masks instead of bounding boxes, allowing for more precise measurement of causal relationships. Additionally, we have modified the GradCAM method to automate the extraction of binary segmentation masks, making it easier to obtain more consistent causal measurements than with manual segmentations and facilitating the use of our method by reducing the dependence on human annotations. The results indicate that the PEC method allows for a more informed assessment of whether a model's predictions and explanations are derived from discernible causal relationships or not. This indicates a promising direction for improving the level of understanding and confidence we can gain by using DL models as tools for Computer-Aided Diagnostics (CADx).

Keywords: Computer-aided diagnosis (CADx), explainable AI (XAI), deep learning (DL), medical image analysis, morpho-constitutional analysis (AMC), kidney stones.

1. Introducción

La identificación temprana del tipo de piedra renal depende de la necesidad de dicha clasificación por parte de un urólogo para determinar e iniciar el tratamiento. Además, varios países desarrollados, como señalan [8] y [6], informan de una incidencia significativa de litiasis urinaria (formación o presencia de cálculos renales), con alrededor de un 10 % de su población experimentando un episodio de cálculos renales al menos una vez en su vida. Además, hay una tasa de recurrencia notablemente alta del 40 % en estos países. Comúnmente, el procedimiento de análisis y clasificación de los cálculos renales, conocido como Análisis Morfo-Constitucional (AMC) [2] es largo, caro y requiere una gran experiencia.

Además, se ha demostrado que el análisis de imágenes médicas, así como el AMC, dependen en gran medida del operador [3, 18, 20]. Además de estas dificultades, debido al creciente número de pacientes cada año y a la gran diversidad natural de casos

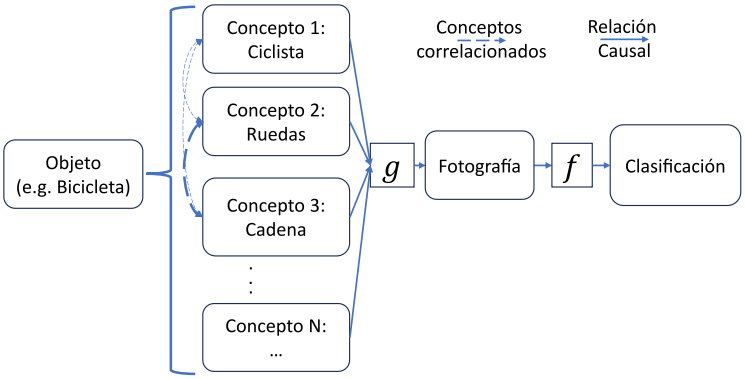


Fig. 1. Gráfico causal que relaciona instancias (e.g. bicicleta), sus conceptos de alto nivel (e.g. ciclista, ruedas, cadena etc.), con la imagen resultante (fotografía) y un clasificador de Deep Learning f, que da una clasificación. El borde discontinuo indica conceptos correlacionados, independientemente de su tipo de relación. Las aristas que conectan los conceptos con la imagen, a través de g, corresponden al proceso natural de generación de imágenes (e.g una cámara al tomar una fotografía).

médicos, el ámbito de la medicina necesita constantemente métodos más precisos y rápidos [17]. Los recientes avances en el campo de la IA, específicamente los avances en Deep Learning (DL) han impulsado la adopción temprana de modelos DL en imágenes médicas [9, 7]. En el contexto de la AMC, se han propuesto métodos desarrollados para el reconocimiento in-vivo o ex-vivo de cálculos renales [12, 10, 5, 19]. Aunque la mayoría de los métodos basados en DL superan a los no basados en DL en términos de precisión, carecen de capacidad de explicación, ya que los modelos DL simplemente emiten una clasificación para una entrada, independientemente de si es por las razones adecuadas o no.

Sin embargo, dado que el campo del análisis de imágenes médicas conlleva decisiones de alto riesgo, principalmente el diagnóstico, que tiene un impacto directo y profundo en la vida de los pacientes, no se puede exagerar la necesidad de un análisis automatizado robusto de imágenes médicas para el Computer-Aided Diagnosis (CADx) [15]. Por lo tanto, los especialistas médicos necesitan entender cómo las características de la imagen de entrada causaron la salida del modelo DL [4]. Precisamente, el campo de la eXplainable AI (XAI) busca proporcionar una comprensión del comportamiento de un modelo DL.

Bajo este objetivo principal, la mayoría de los métodos XAI propuestos relacionados a clasificación de imágenes destacan las causas de la salida de un modelo a partir de su entrada [1], es decir, las explicaciones al señalar la parte de la imagen de entrada que se cálculo causa la salida del modelo DL f se espera refleje el proceso inverso al natural de generación g de la imagen, representado en la Fig.1.

Sin embargo, esta relación causal presuntamente presente en las explicaciones se ha dejado sin una medida cuantitativa. Por ello, para abordar dichas carencias, en este trabajo hemos I) adaptado un método [11], en un conjunto de datos ex-vivo de piedras renales [2], para la puntuación causal, de la relación entre las características latentes en un clasificador de imágenes y su clasificación de salida.

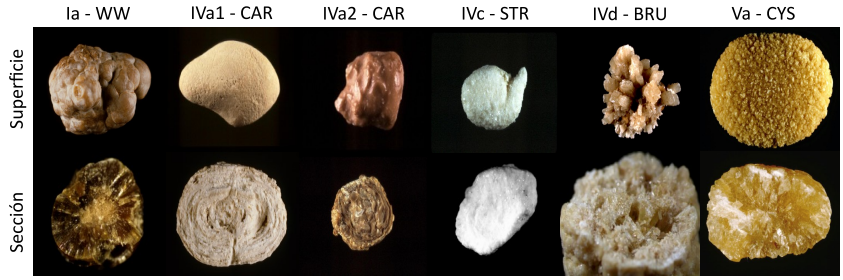


Fig. 2. Ejemplos de los seis subtipos diferentes de cálculos renales del conjunto de datos [2]. El nombre completo de las piedras renales son Whewellite (Ia - WW), Carbapatite (IVa1 & IVa2, CAR), Struvite (IVc - STR), Brushite (IVd -BRU), y Cystine (Va - CYS). Aquí mostramos ejemplos de ambos tipos de vistas en el conjunto de datos por clase, vista de la "Superficie" de las piedras, y vista de la"Sección" transversal.

La adaptación propuesta permite una medición más precisa puesto que trabaja con máscaras de segmentación, en lugar de cajas delimitadoras como lo hacen trabajos previos [11]. Adicionalmente, nuestra adaptación produce una medida causal entre 0 y 1, en lugar de cualquier valor positivo. También, II) modificamos un método de IA explicable (Grad-CAM [16]), para automatizar la extracción de máscaras de segmentación binarias de la región de mayor interés para el modelo, permitiendo obtener medidas causales más consistentes que con máscaras de segmentación anotadas por humanos. Con nuestro método: Puntuación de Explicación Causal (PEC), proporcionamos una forma de validar causalmente las salidas de un modelo DL basado en las áreas de la imagen de entrada indicadas por las explicaciones generadas para la misma entrada y salida del modelo.

Y lo que es más importante, mostramos resultados que indican que nuestro método (PEC) obtiene mejores resultados que cuando se utilizan máscaras de segmentación de los objetos de interés anotadas por humanos. Así pues, nuestro trabajo se enfoca en permitir que los especialistas de la salud aprovechen los hallazgos de los modelos DL para el diagnóstico, comprendiendo la lógica que subyace a dichos resultados, al tiempo que se facilita la aplicación responsable de estas potentes tecnologías de IA en el diagnóstico médico, logrando un equilibrio crucial entre la eficiencia de las máquinas y la responsabilidad humana.

2. Conjunto de datos y métodos

2.1. Conjunto de datos de piedras renales

Nuestro conjunto de datos ex-vivo, Fig.2, se divide en 209 imágenes de superficie y 157 de sección, que en total suman 366 imágenes. Estas imágenes se adquirieron con una cámara digital (CCD) en condiciones de iluminación controladas y con un fondo uniforme. El conjunto de datos está clasificado por los subtipos de cálculos renales, seis en nuestro caso. Estos subtipos, como se muestran en la Fig.2, son la Whewellite, subtipo Ia (Ia - WW), Carbapatite subtipo IVa1 (IVa1 - CAR), Carbapatite subtipo IVa2 (IVa2 - CAR), Struvite subtipo IVc (IVc - STR), Brushite subtipo IVd (IVd - BRU) y Cystine subtipo Va (Va - CYS).

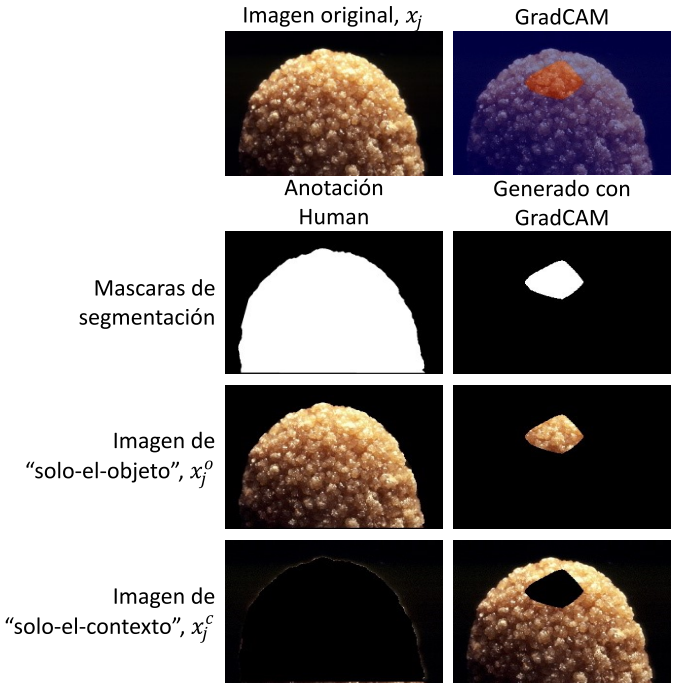
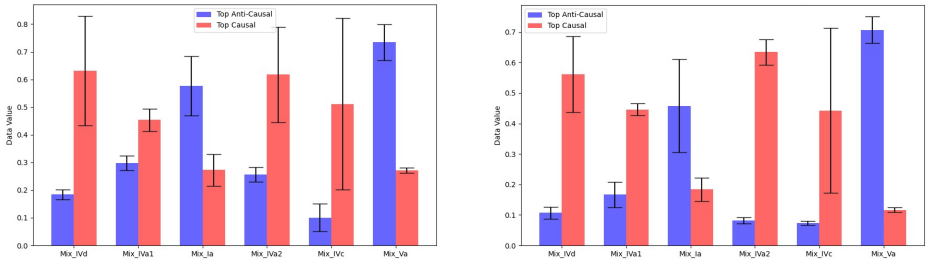


Fig. 3. Arriba a la izquierda: una imagen de ejemplo x_j del conjunto de datos. Arriba a la derecha: Grad-CAM para la clase predicha correspondiente, con umbral para mantener el 30 % de los valores más altos, en rojo. A partir de las segmentaciones del conjunto de datos anotadas por humanos y de las segmentaciones Grad-CAM con umbral, se obtuvieron las imágenes de "solo-el-objeto" x_j^o y "solo-el-contexto" x_j^c.

2.2. Método: Puntuación de explicación causal (PEC)

Inspirándonos en [11], modificamos ligeramente su propuesta "Feature Ratio (FR)" [11] para comprobar la relevancia de las características causales/anticausales señaladas por una segunda red denominada "Neural Causation Coefficient (NCC)". Nuestra modificación es la transformación de las puntuaciones FR originales para que estén acotadas entre 0 y 1, como se ve en la Ec.2, para facilitar su comparación entre diferentes FRs. El modelo NCC es un clasificador binario que indica si la activación de una característica de la última capa convolucional de una CNN se considera causal o anticausalmente relacionada con la salida del modelo.

Para establecer resultados de referencia fácilmente comparables en este trabajo y para futuras implementaciones, empleamos ResNet18, el cual es el modelo utilizado como función f en la Fig. 1. El modelo de ResNet18 f es una red neuronal convolucional que se ha utilizado ampliamente en este campo como clasificador sobre el cual el modelo NCC evaluará las puntuaciones causales. Las puntuaciones causales se obtienen de cada una de las 512 características de activación, $f_l \in \mathbb{R}^{512}$, ya que este es el tamaño de salida de la ultima capa convolucional de la ResNet18 f, considerada como el resultado del extractor de características del modelo.



(a) Puntajes de "Objeto" PEC en base a mascaras anotadas por humanos.

(b) Puntajes de "Objeto" PEC en base a mascaras automáticas con Grad-CAM.

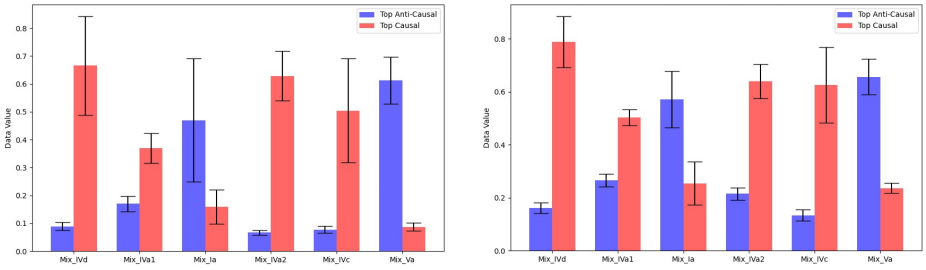
Fig. 4. Puntuación de Explicación Causal (PEC) para las imágenes de "Object-only". En 4a se utilizaron máscaras anotadas manualmente y en 4b se obtuvieron máscaras a partir del método Grad-CAM adaptado. En ambos casos se obtuvo la puntuación PEC para el "objeto".

La arquitectura NCC se entrena siguiendo el trabajo previo [11], utilizando los hiperparámetros y configuración de la implementación en [14]. El conjunto de datos utilizado para la prueba fue el conjunto de datos Tübingen, versión 1.0 [13], que es una colección de ciento ocho muestras observacionales causa-efecto del mundo real. Ya que el conjunto de datos Tübingen se utiliza habitualmente como referencia estándar en el campo de la inferencia causal. El modelo NCC entrenado obtuvo una precisión del 72 %, en lugar del 79 % comunicado anteriormente, a la hora de clasificar las relaciones causales y anticausales.

Aunque esta diferencia pone de manifiesto un área de mejora, es útil para nuestro objetivo actual de demostrar si una puntuación causal es más precisa cuando el "objeto de interés" se identifica mediante un método explicable, en lugar de ser anotado por humanos. Los mapas de características obtenidos de las m imágenes de entrada x_j y salida y_j, por clase $k \in \{1, \ldots, 6\}$, se procesan como un conjunto de pares de entrada (X, Y) por la red NCC. De esta manera, obtenemos las puntuaciones de NCC para cada mapa de características de cada imagen de entrada y las promediamos por mapa de características.

Para cada categoría k y el top 1 % de las característica f_l de acuerdo a sus puntuaciones causales y anticausales, determinaremos su relevancia como una característica de objeto f_l^o o característica de contexto f_l^c. Para ello, preparamos dos versiones alternativas de cada imagen de entrada x_j, las imágenes "solo-el-objeto" x_j^o y "solo-el-contexto" x_j^c. La imagen "solo-el-objeto" x_j^o, indicado por el área blanca de la máscara de segmentación, denota la sección de la imagen original que contiene el "objeto" correspondiente a la clase de la imagen x_j.

De forma complementaria, el recorte "solo-el-contexto" x_j^c, es el área negra de la máscara de segmentación e indica el "contexto", que podemos considerar el fondo de la imagen de entrada x_j. Un ejemplo de ambas imágenes, "solo-el-objeto" x_j^o y "solo-el-contexto" x_j^c, para ambos tipos de segmentación (anotada manualmente y Grad-CAM) pueden verse en la Fig.3. Las características f_l con puntuaciones causales y anticausales en el top 1 % más alto, para los mapas de características promediados por clase k, se seleccionan para informar de sus Feature Ratios (FRs) de acuerdo con la Ec. 1.



(a) Puntajes de "Contexto" PEC en base a mascaras anotadas por humanos

(b) Puntajes de "Contexto" PEC en base a mascaras automáticas con Grad-CAM

Fig. 5. Puntuación de Explicación Causal (PEC) para las imágenes de "solo-el-contexto". En (a) se utilizaron máscaras anotadas manualmente y en (b) se obtuvieron máscaras a partir del método Grad-CAM adaptado. En ambos casos se obtuvo la puntuación PEC para el "contexto".

Estos FRs nos permiten determinar en qué medida cada característica f_l es imputable a la máscara de segmentación del objeto (ratio de la característica del objeto, s_j^o) de la categoría k, o del contexto (ratio de la característica del contexto, s_j^c):

$$s_j^o = \frac{\sum_{j=1}^{m} |f_{jl}^c - f_{jl}|}{\sum_{j=1}^{m} |f_{jl}|}, \qquad s_j^c = \frac{\sum_{j=1}^{m} |f_{jl}^o - f_{jl}|}{\sum_{j=1}^{m} |f_{jl}|}. \tag{1}$$

Adaptación de Feature Ratio (FR): En esta propuesta, los ratios anteriores de "Objeto" s_j^o y "Contexto" s_j^c se adaptan para ser transformadas en valores positivos, acotados entre 0 y 1, en la siguiente Ec.2:

$$\sigma_o(s_j^o) = \frac{2}{1 + e^{-s_j^o}} - 1, \qquad \sigma_c(s_j^c) = \frac{2}{1 + e^{-s_j^c}} - 1. \tag{2}$$

La ResNet18 utilizada como clasificador se entrenó con el conjunto de datos de piedras renales descrito en la Sec.2.1. El entrenamiento consistió en 30 épocas, utilizando el optimizador Adam, con una tasa de aprendizaje de 0,0001. Esta ResNet18 tiene dos capas totalmente conectadas con 512 neuronas y la capa de salida final tiene 6 neuronas para la clasificación de las 6 clases de cálculos renales. De este modelo clasificador ResNet18 f, se utilizaron sus capas convolucionales como extractor de características.

Máscaras de segmentación anotadas manualmente: Las imágenes de "solo-el-objeto" y "solo-el-contexto" de los cálculos renales se obtuvieron a partir de máscaras de segmentación anotadas manualmente con valores 0 o 1 para el contexto y el objeto en la imagen respectivamente. La imagen de "solo-el-objeto" se obtiene multiplicando la imagen de entrada original y su correspondiente máscara de segmentación. A continuación, restamos de la imagen original la imagen "solo-el-objeto", lo que da como resultado la imagen "solo-el-contexto".

Máscaras de segmentación con Grad-CAM: Grad-CAM se caracteriza por calcular un mapa de calor, a partir de las activaciones de un modelo DL y sus gradientes. Sin embargo, las explicaciones producidas por Grad-CAM pueden llegar a tener todos sus valores iguales a cero para algunas entradas, para remediar esta situación aplicamos la modificación de elevar al cuadrado los elementos de la matriz de saliencia de Grad-CAM en lugar del paso de activación con la función ReLU [16]. Con ello se pretende mantener en el mapa de calor de Grad-CAM los valores más salientes, independientemente de su signo original.

Los mapas de Grad-CAM indican el área más relevante de la imagen de entrada para su correspondiente clasificación [16]. En estos mapas de calor se utilizó un umbral, para retener el 30 % de las activaciones más altas en el mapa de calor (el área más importante) como la porción "solo-el-objeto", con valores de 1, y el área restante, considerada menos importante, como la porción "solo-el-contexto", con valores de 0. De esta manera, obtuvimos máscaras de segmentación a partir de Grad-CAM. Este proceso se repite para todas las imágenes del conjunto de datos, con lo que obtenemos un conjunto de segmentaciones generadas por Grad-CAM.

Finalmente, se aplica el mismo proceso para las "Máscaras de segmentación anotadas manualmente" utilizando las segmentaciones generadas por Grad-CAM para obtener sus correspondientes imágenes "solo-el-objeto" y "solo-el-contexto", un ejemplo de los resultados obtenidos se puede ver en la Fig.3. La modificación de los FR "s" en la Ec.2, y el uso de máscaras de segmentación obtenidas automáticamente a partir del Grad-CAM adaptado, es nuestro método de Puntuación de Explicación Causal (PEC) propuesto.

3. Resultados y discusiones

Como se observa en la Fig.4 y la Fig.5, las mediciones causales/anticausales basadas en máscaras anotadas manualmente y las explicaciones de mapas de calor son posibles, incluso con menos varianza que con las máscaras de segmentación anotadas manualmente, que requieren mucho tiempo. Además, los resultados obtenidos entre las máscaras de segmentación anotadas manualmente Fig. 4a y los resultados de las máscaras generadas con segmentación Grad-CAM, Fig.4b fueron notablemente similares en los valores de sus medias, así como también para los resultados de "solo-el-contexto" en Fig. 5a y Fig. 5b. En los resultados, la máscara de segmentación Grad-CAM presentan la ventaja de una menor varianza que los resultados de las anotaciones manuales que recortan la piedra completa para las puntuaciones "solo-el-objeto".

Limitaciones: Ni el trabajo previo utilizado como inspiración [11], ni nuestra propuesta PEC hasta el momento consideran el caso para la identificación de correlaciones entre pares de características de activación f_l del modelo clasificador f. Es necesario realizar mediciones adicionales con diferentes Redes Neuronales Convolucionales (CNNs) para analizar que tan consistentes son los resultados. El uso de un valor umbral arbitrario podría estar limitando los resultados obtenidos con las máscaras de segmentación Grad-CAM. Un hallazgo clave es que para la mayoría de las clases, 4 de 6, la puntuación causal FR de "objecto" es predominante, como se muestra

en Fig.4, contradictoriamente al hallazgo en [11]. Esta diferencia clave puede deberse a la menor puntuación de rendimiento de NCC para la clasificación de señales causales, y a la limitada cantidad de muestras de datos en el conjunto de datos, de solo 366 para las 6 clases.

4. Conclusiones y trabajo futuro

Las mediciones causales basadas en las características más relevantes de la imagen de entrada son favorables. Nuestro método, PEC, demuestra que es posible automatizar las mediciones causales teniendo acceso a los pesos del modelo DL. Además, con nuestra propuesta, PEC, es posible dar a los especialistas una indicación de qué características de un modelo son las más relevantes y si éstas guardan una relación causal o anticausal con la salida. No obstante, son necesarios más experimentos.

Como trabajo futuro, deberían explorarse diferentes niveles de umbrales de segmentación para identificar el valor óptimo para evaluar tanto las puntuaciones causales como las anticausales. Para nuestro conjunto de datos, en particular, esto es relevante, debido a que las grandes áreas originales de fondo negro y las máscaras de segmentación obtenidas de Grad-CAM son empíricamente pequeñas, como se observa en Fig. 3.

Una dirección interesante a explorar para mejoras es modificar las puntuaciones FR de “Objeto” y “Contexto” para que se basen en un enfoque de aprendizaje métrico (metric learning) sobre el espacio latente extraído por las capas convolucionales del clasificador de imágenes, en lugar del cambio de activación de la imagen original frente a los recortes de “Objeto” o “Contexto” únicamente. Por último, igualmente la aplicación de distintos métodos XAI para obtener las máscaras de segmentación se deja para futuros trabajos.

Acknowledgments. The authors wish to acknowledge the Mexican Council for Science and Technology (CONAHCYT) for the support in terms of postgraduate scholarships in this project, and the Data Science Hub at Tecnologico de Monterrey for their support on this project. This work has been supported by Azure Sponsorship credits granted by Microsoft’s AI for Good Research Lab through the AI for Health program. The project was also supported by the French-Mexican ANUIES CONAHCYT Ecos Nord grant 322537.

Compliance with ethical approval. The images were captured in medical procedures following the ethical principles outlined in the Helsinki Declaration of 1975, as revised in 2000, with the consent of the patients.

Referencias

1. Borys, K., Schmitt, Y. A., Nauta, M., Seifert, C., Krämer, N., Friedrich, C. M., Nensa, F.: Explainable AI in medical imaging: An overview for clinical practitioners – Beyond saliency-based XAI approaches. European Journal of Radiology, vol. 162, pp. 110786 (2023) doi: 10.1016/j.ejrad.2023.110786

2. Corrales, M., Doizi, S., Barghouthy, Y., Traxer, O., Daudon, M.: Classification of stones according to Michel Daudon: A narrative review. European Urology Focus, vol. 7, no. 1, pp. 13–21 (2021) doi: 10.1016/j.euf.2020.11.004
3. De-Coninck, V., Keller, E. X., Traxer, O.: Metabolic evaluation: Who, when and how often. Current Opinion in Urology, vol. 29, no. 1, pp. 52–64 (2019) doi: 10.1097/mou.0000000000000562
4. Flores-Araiza, D., Lopez-Tiro, F., El-Beze, J., Hubert, J., Gonzalez-Mendoza, M., Ochoa-Ruiz, G., Daul, C.: Deep prototypical-parts ease morphological kidney stone identification and are competitively robust to photometric perturbations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 295–304 (2023) doi: 10.48550/arXiv.2304.04077
5. Gonzalez-Zapata, J., Lopez-Tiro, F., Villalvazo-Avila, E., Flores-Araiza, D., Hubert, J., Ochoa-Ruiz, G., Daul, C., Mendez-Vazquez, A.: A metric learning approach for endoscopic kidney stone identification. Expert Systems with Applications, vol. 255, pp. 124711 (2024) doi: 10.1016/j.eswa.2024.124711
6. Hall, P. M.: Nephrolithiasis: Treatment, causes, and prevention. Cleveland Clinic Journal of Medicine, vol. 76, no. 10, pp. 583–591 (2009) doi: 10.3949/ccjm.76a.09043
7. Jiang, H., Diao, Z., Shi, T., Zhou, Y., Wang, F., Hu, W., Zhu, X., Luo, S., Tong, G., Yao, Y. D.: A review of deep learning-based multiple-lesion recognition from medical images: classification, detection and segmentation. Computers in Biology and Medicine, vol. 157, pp. 106726 (2023) doi: 10.1016/j.compbiomed.2023.106726
8. Kasidas, G. P., Samuell, C. T., Weir, T. B.: Renal stone analysis: Why and how? Annals of Clinical Biochemistry: International Journal of Laboratory Medicine, vol. 41, no. 2, pp. 91–97 (2004) doi: 10.1258/000456304322879962
9. Lee, L. I. T., Kanthasamy, S., Ayyalaraju, R. S., Ganatra, R.: The current state of artificial intelligence in medical imaging and nuclear medicine. BJR—Open, vol. 1, no. 1, pp. 20190037 (2019) doi: 10.1259/bjro.20190037
10. Lopez, F., Varelo, A., Hinojosa, O., Mendez, M., Trinh, D. H., ElBeze, Y., Hubert, J., Estrade, V., Gonzalez, M., Ochoa, G., Daul, C.: Assessing deep learning methods for the identification of kidney stones in endoscopic images. In: Proceedings of the 43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 2778–2781 (2021) doi: 10.1109/EMBC46164.2021.9630211
11. Lopez-Paz, D., Nishihara, R., Chintala, S., Scholkopf, B., Bottou, L.: Discovering causal signals in images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) doi: 10.48550/arXiv.1605.08179
12. Lopez-Tiro, F., Estrade, V., Hubert, J., Flores-Araiza, D., Gonzalez-Mendoza, M., Ochoa, G., Daul, C.: On the in vivo recognition of kidney stones using machine learning. IEEE Access, vol. 12, pp. 10736–10759 (2024) doi: 10.1109/access.2024.3351178
13. Mooij, J. M., Peters, J., Janzing, D., Zscheischler, J., Schölkopf, B.: Distinguishing cause from effect using observational data: Methods and benchmarks. Journal of Machine Learning Research, vol. 17, no. 32, pp. 1103–1204 (2016)
14. Park, S.: Neural-causation-coefficient (2023) github.com/euphoria0-0/Neural-Causation-Coefficient
15. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence, vol. 1, no. 5, pp. 206–215 (2019) doi: 10.1038/s42256-019-0048-x
16. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: IEEE International Conference on Computer Vision (2017) doi: 10.1109/iccv.2017.74
17. Topol, E.: Deep medicine: How artificial intelligence can make healthcare human again. Basic Books (2019)

18. Zhu, C., Doyle, T. E., Noseworthy, M. D.: Ultrasound operator variance classification for agency in artificial intelligence support of cyber-physical systems. In: IEEE Canadian Conference on Electrical and Computer Engineering, IEEE, pp. 446–451 (2022) doi: 10.1109/ccece49351.2022.9918266
19. Zhu, W., Zhou, R., Yuan, Y., Timothy, C., Jain, R., Luo, J.: Segprompt: Using segmentation map as a better prompt to finetune deep models for kidney stone classification (2023) doi: 10.48550/arXiv.2303.08303
20. Åkesson, L., Svensson, A., Edenbrandt, L.: Operator dependent variability in quantitative analysis of myocardial perfusion images. Clinical Physiology and Functional Imaging, vol. 24, no. 6, pp. 374–379 (2004) doi: 10.1111/j.1475-097x.2004.00574.x

ISSN 1870-4069

Detección automática de averías en el pavimento mediante visión por computadora y cómputo móvil

María de Jesús Galindo-López, Arturo Salazar-Segundo, José Alberto Hernández-Aguilar

Universidad Autónoma el Estado de Morelos, Facultad de Contaduría, Administración e Informática, México

{mariadejesus, arturo.salazar, jose_hernandez}@uaem.mx

Resumen. En esta investigación se discute la detección automática de averías en el pavimento mediante visión por computadora. La detección de éstas se realiza combinando tecnologías computacionales y móviles, lo que puede ser crucial para el mantenimiento y seguridad de las vialidades de México, ya que son vistas como uno de los principales problemas en las ciudades. En esta investigación se discute las imperfecciones del pavimento previas a que se conviertan en baches, que pueden resultar peligrosos sobre todo de noche y en temporada de lluvias, ya que el agua estancada camufla la dimensión del bache, provocando daños que van de leves a graves no solo en los vehículos, sino que pueden llegar a causar accidentes con consecuencias desastrosas. Metodología: se recopilaron imágenes de imperfecciones en el asfalto utilizando teléfonos inteligentes, las cuales son procesadas para identificar sus características como forma, latitud y longitud, las cuales se procesan mediante visión por computadora utilizando Python, Scikit-learn, Opencv, y Tensorflow. En esta investigación se utilizó el modelo preentrenado resnet152 para identificar averías en el pavimento y se probó con el conjunto de datos propio. Los resultados preliminares indican que se tiene una precisión del 70 % para identificar averías en el pavimento.

Palabras clave: Detección de averías, pavimento, visión por computadora, cómputo móvil.

Automatic Pavement Fault Detection Using Computer Vision and Mobile Computing

Abstract. This research discusses the automatic detection of pavement defects using computer vision. Detecting these defects is done by combining computer and mobile technologies, which can be crucial for the maintenance and safety of Mexico's roads since they are seen as one of the main problems in cities. This research discusses the imperfections in the pavement before they become potholes, which can be dangerous, especially at night and during the rainy season, since stagnant water camouflages the size of the pothole, causing damage ranging from minor to severe not only to vehicles but can also cause accidents with disastrous consequences. Methodology: Images of asphalt imperfections were collected using smartphones, which are processed to identify their characteristics,

such as shape, latitude, and longitude, which are processed through computer vision using Python, Scikit-learn, Opencv, and Tensorflow. This research used the pre-trained model resnet152 to identify pavement damage and was tested with the in-house dataset. Preliminary results indicate a 70% accuracy in identifying pavement damage.

Keywords: Fault detection, pavement, computer vision, mobile computing.

1. Introducción

1.1. Antecedentes

Los baches se han convertido en uno de los problemas más comunes en las calles de México al grado de que la población los considera un foco de inseguridad vial, el 81 % de la población de 18 años y más, manifiestó que los baches en calles y avenidas son uno de los problemas más importantes en su ciudad de acuerdo al Instituto Nacional de Estadística y Geografía en México [11]. Autores como Dhiman y Kettle (2019) [8] , y Guo y Zhang (2022) [10] , discuten el uso de diferentes algoritmos para la detección de anomalías basados en los datos de modelos pre-entrenados a fin de detectar grietas, escamas o baches. En Azar et al. (2019) [6] se discute la combinación de estas técnicas con el Internet de las cosas para ese mismo propósito.

La Inteligencia Artificial de las Cosas (del inglés Artificial Intelligence of Things - AIoT), se define como una extensión del internet hacia el ámbito físico, mientras que otros lo describen como un colectivo de sensores colocados en “cosas” y dentro de infraestructuras cibernéticas de acuerdo a Azar et al. (2019) [6]. En ella se unen las capacidades de IoT (Internet of Things), Big data e Inteligencia artificial, todas necesarias para la identificación de averías en el pavimento.

1.2. Problema de investigación

La principal línea de investigación de este proyecto se concentra en el desarrollo de un sistema que permita la detección y geolocalización de averías (anomalías) en el pavimento mediante visión por computadora y cómputo móvil.

1.3. Justificación

En Azar et al. (2020) [5] se describe una creciente demanda a nivel global de productos y servicios tanto en el campo Internet de las cosas como en el de Inteligencia Artificial (IA), tan solo para el año 2020 se esperaban inversiones de hasta 1.5 billones de dólares . Estas tecnologías hacen posible el desarrollo de sistemas que permiten la identificación de averías en el pavimento de manera muy precisa. Este problema no solo lo enfrentan países latinoamericanos como México sino también países como India, Turquía, Corea del Sur, Japón por mencionar algunos.

Fig. 1. Ejemplo de avería tipo piel de cocodrilo en asfalto, en esta figura se muestra un conjunto de grietas acumuladas en el pavimento que asemejan la piel de reptil.

1.4. Objetivo

Diseñar un sistema adaptable a automóviles que permita detectar averías en el pavimento de calles y ciudades mediante visión por computadora, el desarrollo de algoritmos inteligentes, y el uso de sensores cámaras y GPS (del inglés Global Position System - Sistema de Posicionamiento Global) en entornos de IoT mediante dispositivos móviles.

1.5. Hipótesis

Ho. Es posible la implementación de un sistema de detección de averías en el pavimento mediante el análisis de imágenes por visión por computadora y las tecnologías móviles actuales.

1.6. Contribución

La principal contribución de esta investigación es: la creación de un sistema que permita prevenir la formación de baches en el Estado de Morelos, mediante la detección de imperfecciones (averías) en la carpeta asfáltica mediante visión por computadora y cómputo móvil, así mismo se aporta una base de datos real de averías en el asfalto en el Estado de Morelos. Lo que proporciona información valiosa que permite la planificación del mantenimiento preventivo y correctivo, pudiendo reducir los costos de reparaciones tardías, contribuyendo con ello a la mejora de la seguridad vial.

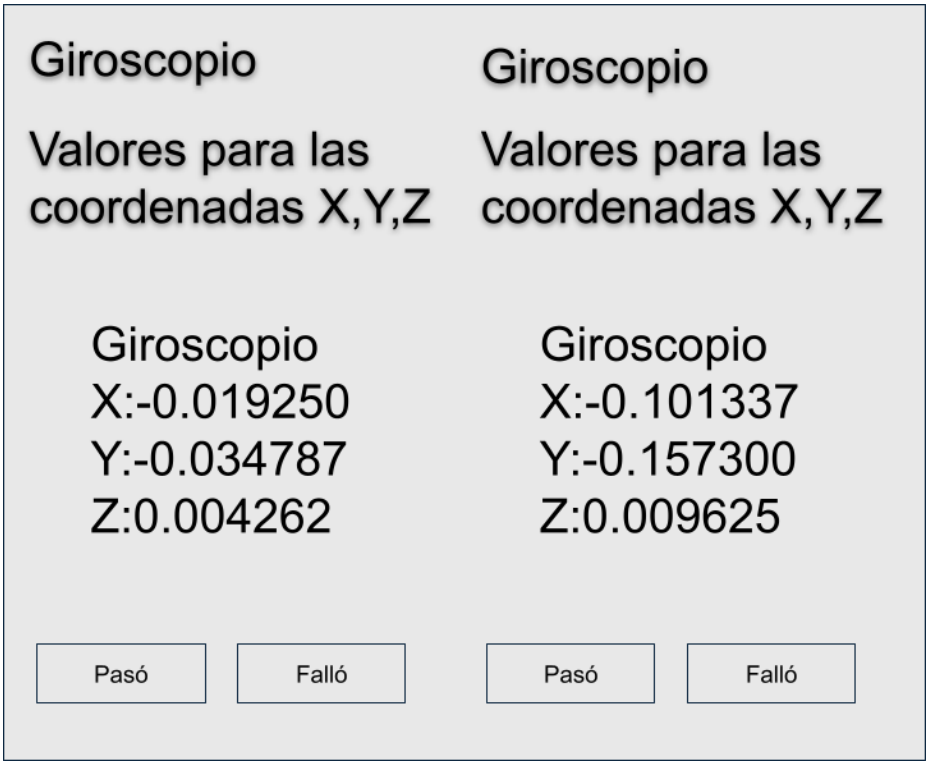


Fig. 2. Detalle de lecturas de los sensores de giroscopio de un teléfono inteligente. En la pantalla se muestran las lecturas en un punto en particular, basado en [7].

1.7. Alcances y limitaciones

Se utilizará el modelo preentrenado resnet152_rdd_19_best8140_infer para identificar las averías en el pavimento mediante transferencia de aprendizaje (transfer learning). Para la etapa de prueba se utilizará un conjunto de datos propio, que consta de 75 imágenes de averías en el pavimento registradas con dispositivos móviles en el Estado de Morelos. Los principales desafíos técnicos son: la posición de la cámara, la calidad de la imágen, la velocidad del vehículo en el que va montada la cámara, y las condiciones meteorológicas en las que se pruebe el desempeño del modelo.

1.8. Estructura del documento

En la primera sección se discute la problemática que se quiere resolver, el objetivo general, hipótesis y principal contribución. En la segunda sección se analiza el trabajo relacionado. En la tercera sección se presenta la metodología propuesta. En la cuarta sección se discuten los resultados obtenidos. En la quinta sección se presentan conclusiones y trabajos futuros. Finalmente, se listan las referencias utilizadas.

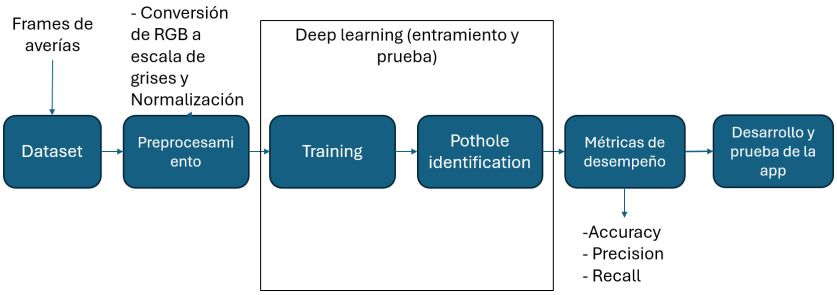


Fig. 3. Metodología propuesta (Fuente propia).

2. Trabajo relacionado

Según Manzanares-González (2019) [13] existen diferentes tipos de averías e imperfecciones en las carreteras y calles de México, estás pueden ir desde simples grietas - ya sean transversales de corto, mediano y largo alcance-, las que ocurren por fatiga del asfalto también llamadas piel de cocodrilo, hasta hoyos también llamados baches. Los desperfectos mostrados en la (Fig. 1) son una clara señal de un futuro bache, que dependiendo del tamaño de la imperfección, el tamaño y la forma del bache podrían variar, aunque la profundidad dependerá de diversos factores como el clima, nivel de tránsito vehicular, tiempo de la avería, y la temporada del año, entre los más comunes.

2.1. Tipos de pavimentación

De acuerdo Sandstone Global (2024), en México existen diversos tipos de imperfecciones en las calles y carreteras. La pavimentación consiste en la colocación de capas de materiales en el suelo o nivel superior de la terracería, a la que posteriormente se le aplica el asfalto, losa, piedra, ladrillos, concreto, concreto hidráulico, por mencionar algunos, que conformará la superficie de rodamiento, los tipos de pavimento mas utilizados son:

Pavimento flexible. Tiene como superficie de rodadura una capa de mezcla asfáltica comúnmente llamada asfalto. Las cargas de los vehículos hacia las capas inferiores se distribuyen mediante las características de fricción y cohesión de las partículas de los materiales, mientras que la carpeta asfáltica se pliega a pequeñas deformaciones de las capas inferiores sin que su estructura se rompa.

Pavimento semirrígido. La carpeta asfáltica se apoya en una base asfáltica o en una base estabilizada con cemento Portland. Este material es un tipo de cemento hidráulico compuesto principalmente de clinker -mezcla de roca caliza y arcilla- más yeso. Comúnmente se llama pavimento tipo adoquín.

Pavimento rígido. Tiene como superficie de rodadura una losa de concreto apoyada en capas de diversos materiales. Este tipo de pavimento no puede plegarse a las deformaciones de las capas inferiores sin que se presente una falla estructural. Este tipo de pavimento se caracteriza por tener en su capa superior concreto hidráulico, y es el que se empleó en el paso exprés de la ciudad de Cuernavaca Morelos. En este trabajo nos enfocaremos en el análisis de averías en pavimento flexible y pavimento rígido.

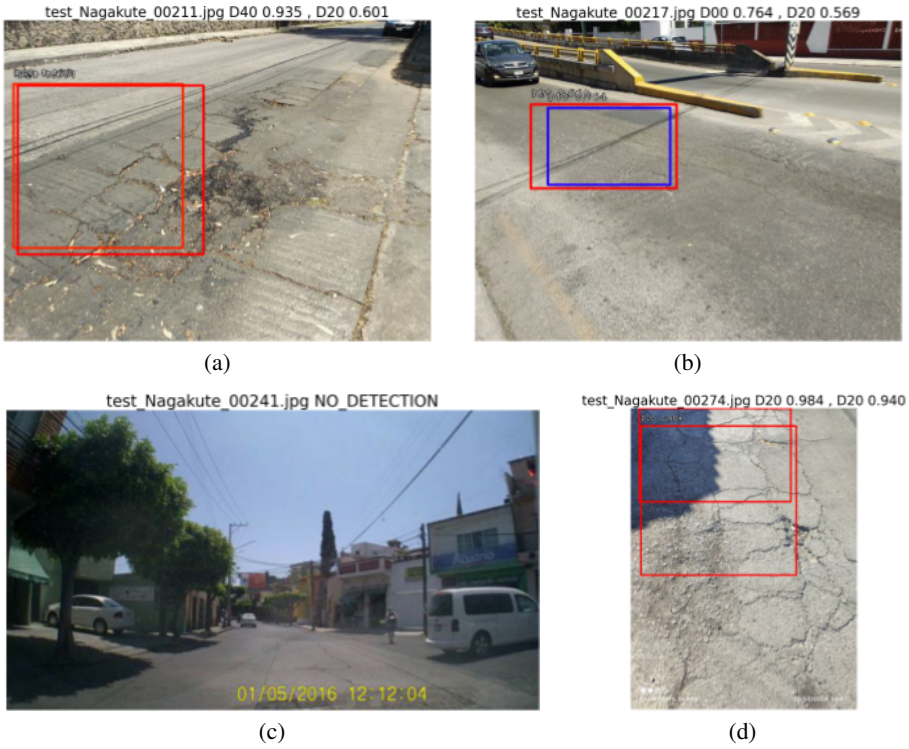


Fig. 4. Predición de averías en diferentes escenarios.

2.2. Detección de averías en el pavimento

Existen proyectos que plantean diferentes enfoques para la detección de baches, algunos utilizan machine learning, otros deep learning, o una combinación de ellos, así como sensores GPS, giroscopios y acelerómetros para recopilar datos; otros utilizan bases de datos preexistentes de anomalías en pavimentos que posteriormente analizan y prueban con sus propios algoritmos, para verificar su funcionamiento en la identificación de averías con métricas como Accuracy, Precision o Recall.

Muchos de los datos que se utilizan en este tipo de estudios se recopilan con teléfonos inteligentes, también llamados smartphones, los cuales cuentan con muchos sensores, desde los más básicos como los que detectan luz ambiental, temperatura ambiental, o proximidad para apagar la pantalla en una llamada y evitar tocar botones de forma accidental, hasta los más complejos como los que detectan el rostro o voz en tiempo real, además incluyen cámaras de alta resolución y GPS útiles para la geolocalización de personas y cosas.

Para la Detección de anomalías en la carpeta asfáltica, Ramírez-Venegas (2017) [14] propuso el uso de sensores de teléfonos inteligentes y dispositivos de diagnóstico a bordo (del inglés OnBoad Diagnostic version II - OBDII), además utilizó redes neuronales artificiales (ANN). Usando 2 métodos de análisis, uno que procesa imágenes 2D y otro que procesa vídeos, se obtuvo una precisión en la detección de averías de aproximadamente el 90 %.

Tabla 1. Número de detecciones en algunas imágenes del conjunto de prueba.

Imagen	D00	D01	D10	D11	D20	D40	D43	D44	D30	N. detec.
00206					0,601	0,935				2
00210					0,999					1
00211		0,626			1,000					2
00212	0,764				0,569					2
00224										0
00236										0
00269	0,882 0,712									2
00272					0,984 0,940					2

Cabe señalar que el puerto OBDII se tiene hoy en día en el 100 % del parque vehicular del mundo. Un módulo OBDII es un dispositivo con un puerto simple que almacena datos en tiempo real de los autos, este módulo se usa con el ELM327, el cual cuenta con un módulo Bluetooth habilitado para emparejarse con un teléfono inteligente según Ashwini, Bhagwat y Sharma (2020) [4]. Todas las marcas ensambladoras de vehículos están obligadas desde el año 1996 a instalarlo en sus autos nuevos.

En Azar y Tapia (2018) [7] se presenta un trabajo que emplea una aplicación móvil, que además del acelerómetro de un teléfono inteligente utiliza el giroscopio para la identificación de baches. A partir de los datos recolectados se entrenan modelos con Máquinas de Soporte Vector - (del inglés Support Vector Machines - SVM) logrando una precisión superior al 90 %. En la Fig. 2 se muestran las lecturas obtenidas en un punto en particular mediante un teléfono móvil.

En Azar et al. (2019) [6] se discute la Inteligencia Artificial de las Cosas “móviles”, se menciona que no se necesita de un equipo de cómputo convencional o portátil más que los sensores conectados a un teléfono inteligente para recopilar los datos. Estos sensores hacen referencia a la cámara y al GPS, los cuales se activan al tomar una fotografía, en donde no solo se registra la imagen tomada con una alta resolución, sino también se registra el lugar donde fue tomada, incluyendo su latitud y longitud.

Los autores señalan que al tratarse de tecnologías móviles que al día de hoy tienen recursos limitados, herramientas como Tensorflow, YOLO (You Only Look Once) v5, y APIs (Applications) de alto nivel son esenciales en el desarrollo de una aplicación para la detección de las averías viales, las cuales deben ser optimizadas para su funcionamiento, haciendo además uso de lenguajes de programación como JAVA y Kotlin (Android), Objective C (IOS), C++ y Python que pueden proporcionar flexibilidad y distintos entornos de desarrollo.

En Manzanares-González (2019) [13] se recopilan datos de teléfonos inteligentes, y se utiliza Deep Learning (red neural profunda) para la detección de anomalías en el pavimento basados en modelos pre-entrenados a fin de detectar grietas, escamas o bien baches directamente. En este trabajo, se realizó una selección de la plataforma tecnológica, la cual permitió implementar una red neural en la que se entrenaron diferentes modelos.

Tabla 2. Resumen de la detección de averías.

Tipo	Número
Imágenes en las que sí se detectó averías	42
Imágenes en las que no se detectó averías	33
Total de imágenes procesadas	75

Con el mejor de estos modelos se creó una app para la detección de baches. En Escobar, Flores y Fernández (2023) [9] se discute la estimación de averías en el pavimento utilizando técnicas de procesamiento digital de imágenes, para lo cual se utilizaron técnicas de transformación de perspectiva, umbrales, y filtros, que permiten estimar el área afectada. Los resultados se evaluaron utilizando el método de intersección sobre unión (IOU); se obtuvieron valores de accuracy de 0.69 para piel de cocodrilo, 0.87 para baches, y 0.79 para grietas.

2.3. Bases de datos para la detección de baches

Con respecto a la detección de objetos basada en Deep Learning y aplicada a vehículos autónomos, en Arriola (2018) [1] se menciona que existen varias bases de datos de baches en distintas condiciones meteorológicas, pero algunos investigadores también se apoyan de imágenes propias, ya que el ángulo de visión de muchas imágenes no es el mismo que se lleva dentro de la cabina del vehículo. De acuerdo a Arriola (2018) [1], para entrenar una inteligencia artificial se necesita una base de datos robusta, en este trabajo la prueba consistió de 5874 imágenes las cuales tenían una resolución de 3680×2760 pixeles, que es una resolución aproximada a 4K.

Para acortar tiempos de entrenamiento es viable usar bases de datos ya hechas que se pueden encontrar en internet. De acuerdo a Arya et al. (2021) [2] una de las bases de datos más utilizada para el entrenamiento, prueba y evaluación de averías en el pavimento es el dataset RDD2020: An Image Dataset for Smartphone-based Road Damage Detection and Classification.

Este conjunto de datos contiene 6336 imágenes de caminos de la India, Japón y República Checa con más de 31000 ejemplos de daños en el piso. El conjunto de datos tiene cuatro categorías de daños: Roturas lineales (D00), roturas transversales (D10), roturas de piel de cocodrilo (D20), y baches (D40); y fue desarrollada para probar métodos basados en deep learning que permitan detectar y clasificar daños en el camino automáticamente. Estas imágenes se capturaron utilizando teléfonos inteligentes montados en el vehículo, lo que hace útil a los municipios y agencias de movilidad desarrollar métodos de monitoreo de bajo costo.

El reto global para la detección de daños en el camino (GRDDC'2020), fue organizado por la IEEE Big Data Cup en 2020, y utilizó el dataset RDD2020 para evaluar los modelos de detección de daños en el camino propuestos por los participantes. Una visión general del evento, puede ser consultada en Arya, Maeda y Sekimoto (2024) [3]. Así mismo, hay repositorios donde además de la base de datos, se encuentran los programas que pueden ser utilizados para identificar las averías en el pavimento y que pueden ser utilizados para fines de comparación, como por ejemplo el github de Khokhar (2023) [12].

3. Metodología

La metodología propuesta para esta investigación consta de 6 etapas véase la Fig. 3.

Base de datos. Para este proyecto se utilizaron 75 imágenes de averías en el pavimento con una resolución de 600 x 600 píxeles. Estas imágenes fueron tomadas por los autores en distintos puntos del Estado de Morelos.

Preprocesamiento. Las imágenes se convirtieron a escala de grises y se normalizaron para disminuir la complejidad de procesamiento.

Entrenamiento y prueba del modelo. Estas etapas se llevaron a cabo utilizando redes de convolución CNN - Convolutional Neural Network, para el entrenamiento se utilizó el modelo preentrenado resnet152_rdd_19_best8140_infer descrito en Manzanares-González (2019) [13], y para la prueba se utilizaron las 75 imágenes preprocesadas. El entrenamiento y la prueba del modelo se llevaron a cabo en una computadora de escritorio ADM Ryzen 5, modelo 2600 de 6 núcleos y 12 hilos a 3.40GHz de velocidad, 16GB de RAM DDR4 a 3200 MHz, una tarjeta gráfica AMD Radeon RX 460 de 4GB con interfaz PCIe 3.0×8.

Configuración del ambiente de trabajo. A continuación, se describen los pasos que se siguieron para configurar el ambiente de trabajo:

1. Se importan las librerías y módulos necesarios para el procesamiento de imágenes y detección de objetos Numpy, Keras, Pandas, Tensorflow, Matplotlib, Cv2 y módulos de keras retinanet para detección de objetos.
2. Se configura la sesión de Tensorflow definiendo la función get_session() para permitir el crecimiento de la memoria en las unidades gráficas de procesamiento GPUs.
3. Se define el modelo de Retinanet pre-entrenado asignado a la variable model_trained desde una ubicación específica usando el modelo resnet152, se carga usando la función load_model().
4. Definimos un diccionario que mapea los índices para etiquetar las zonas por el modelo.
5. Se leen el conjunto de datos prueba testset desde un archivo .CSV (delimitado por comas), el cual contiene la lista de imágenes a procesar, este describe el ancho, altura y el nombre de las imágenes, para ello se utiliza la función pd.read_csv() de pandas.
6. Se realiza un ciclo for para la detección de cada imagen, el número de imágenes se establece en 'tail(n)'.
7. Se carga la imágen usando la función read_image_bgr() desde una ruta específica y se añade el formato de la imagen 'format(img)', además de preparar el ambiente para dibujar el delimitador.
8. Se procesa la imagen con la función 'process_image()' y se almacena un copia en 'draw' para convertir la imagen en color a formato 'BGR' a 'RGB' con la función 'cv2.cvtColor()'.

9. Se procesan las imágenes por la red neuronal de tensor flow "tf" pasando previamente por la función 'preprocess_image()', después se redimensiona la imagen con la función 'resize_image()' para tener un tamaño de 600x600 requerido por el modelo.
10. Se realiza la predicción de objetos en la imagen redimensionada y se lleva a cabo la corrección de cambios de la escala.
11. En caso de no encontrar nada, se imprime 'NO_DETECTION'.
12. Iteramos cada predicción de objetos delimitando una caja, un puntaje de confianza y una etiqueta. Si el puntaje es mayor o igual a 0.5 se dibuja la caja delimitadora.
13. Se crea una visualización de cada imagen con las detecciones y la información de los objetos con ayuda de la biblioteca matplotlib; con la variable 'draw' mostramos las cajas delimitadoras, finalmente mostramos en pantalla el resultado de cada imágen procesada con 'plt.show()'.

Métricas. Las métricas que se utilizaron para evaluar el desempeño del sistema están basadas en la matriz de confusión propuesta en Raschka y Mirjalili (2019) [15]: El accuracy (exactitud) es una métrica para evaluar modelos de clasificación. Informalmente es la fracción de predicciones que el modelo realizó correctamente. De manera formal, el accuracy tiene la siguiente definición [15]:

$$\text{Accuracy} = \frac{\text{No.\_de\_predicciones\_correctas}}{\text{No.\_total\_de \_predicciones}}. \tag{1}$$

Para clasificación binaria, el accuracy también se puede calcular en términos de positivos y negativos de la siguiente manera:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{2}$$

donde $\text{TP} =$ Verdaderos positivos, $\text{TN} =$ Verdaderos negativos, $\text{FP} =$ Falsos positivos y $\text{FN} =$ Falsos negativos. También se utilizarán las métricas de Precisión (Precision) y Recall (Sensibilidad). La precisión responde a la pregunta ¿qué proporción de identificaciones positivas fue correcta [15].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{3}$$

El recall responde a la pregunta ¿Qué proporción de positivos reales se identificó en forma correcta? [15]:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{4}$$

Desarrollo y prueba de la App. El sistema entrenado con el modelo de red convolucional se integra en una app en Android Studio mediante el desarrollo por prototipos. El teléfono inteligente se colocará de manera horizontal en el tablero o sobre el volante, la app activará la cámara del dispositivo, y entrará en modo búsqueda de baches, y conforme vaya avanzando, el vehículo identificará los baches

Tabla 3. Métricas de desempeño.

Tipo	Número
Total de anomalías detectadas	66
Verdaderos positivos (TP)	41
Falsos positivos (FP)	18
Detecciones negativas (FN)	7
Accuracy	0,62
Precision	0,70
Recall	0,85

que vayan apareciendo en el horizonte en color verde, marcando su posición GPS (latitud y longitud) y señalando el porcentaje de precisión de la detección correspondiente. Si aparecen varias averías las mostrará también con sus respectivas métricas y geoposición.

4. Resultados y discusión

Las 75 imágenes de prueba fueron tomadas en diferentes ángulos y alturas. Tomando en cuenta esto, con las imágenes que se obtuvieron con cámaras deportivas tipo GoPro de resolución 720p de calidad baja, el modelo no pudo detectar anomalías, salvo en un caso en el cual se apuntó a un poste -véase la imagen 00233-. A partir de la imagen 00256 las fotografías se tomaron en un ángulo de 90°, es decir, la cámara del teléfono apuntando directamente al suelo, considerando circunstancias como la sombra de cables, postes de luz y teléfono.

Un pequeño conjunto de imágenes (siete) fueron tomadas estando de pie y apuntando la cámara del teléfono en un ángulo de 180°, es decir viendo al horizonte, para poder visualizar banquetas, automóviles, sombras y estructuras de casas, esto con el fin de poder diferenciar entre anomalías y sombras.

A continuación en la Figura 4, se muestran los resultados de procesar diferentes imágenes; en la Figura 4a. se muestra una predicción acertada en la sección D40 con una precisión de 0.935 y en la sección D20 con una precisión de 0.601. En la Figura 4b. se muestra un falso positivo, en el que se confundió una sombra de un cable con una avería. En la figura 4c. se muestra un verdadero negativo, es decir existe una avería pero no se detectó; y en la figura 4d. se muestran dos verdaderos positivos en la sección D20, con precisiones de 0.984 y 0.940 respectivamente.

Los resultados completos de la detección en las imágenes tomadas y procesadas con la metodología propuesta pueden ser consultados en Salazar-Segundo (2024). En la Tabla 1 se muestra un subconjunto de los resultados obtenidos. La columna uno indica el número de imagen; las columnas 2 a la 10, el tipo de anomalía detectada de acuerdo a Manzanares-González (2019) [13]; y la última columna indica el número de detecciones. En la tabla 2 se muestra el resumen de la detección de averías, de las 75 imágenes de prueba en 42 sí detectaron averías, y en 33 no se detectaron averías.

De acuerdo a la tabla 3, se alcanzó un Accuracy (exactitud) de 0.62, una Precision (precisión) de 0.70, y un Recall (sensibilidad) de 0.85, lo cual indica que la predicción es buena, pero se requiere de una cantidad mayor de imágenes para mejorar el desempeño de la propuesta. Estos resultados están por debajo de los obtenidos por Manzanares-González (2019) [13] quien obtuvo un porcentaje de acierto de 0.8945, y un porcentaje de fallo de 0.1055. También están por debajo de los obtenidos por Escobar-Arenas et al. (2023) [9] en donde se obtuvo un accuracy promedio de 0.78.

5. Conclusiones y trabajos futuros

Las agencias de transporte y obras públicas pueden mejorar la condición y operación de sus redes de calles y carreteras implementando un sistema de gestión de mantenimiento de pavimentos, que utilice recopilación de datos (imágenes, latitud y longitud) basada en teléfonos inteligentes y predicción mediante visión por computadora, que permita el apoyo a la toma de decisiones. Dado los resultados obtenidos, se concluye que la hipótesis Ho. Es posible la implementación de un sistema de detección de averías en el pavimento mediante el análisis de imágenes por visión por computadora y las tecnologías móviles actuales, se cumple.

Si bien la calidad de la cámara juega un papel muy importante a la hora de grabar vídeo, existen variables que pueden llegar a generar falsos positivos o que simplemente no se muestren datos al momento de procesar las imágenes, es el caso de imágenes tomadas con reflejos del parabrisas provocados por el sol o la suciedad en el mismo, lo que puede confundirse con una anomalía, situaciones que pueden impedir una lectura correcta de las condiciones del pavimento. Además de que las cámaras de los celulares inteligentes no cuentan con ángulos de visión pronunciados (ojo de pez), y que al momento de procesar las imágenes podrían no ser los adecuados.

Para futuras investigaciones, consideramos sujetar la cámara al cofre o a la defensa del vehículo para tener una mejor visión de la vialidad, sobre todo en horarios donde el sol no proyecta sombras en las averías (medio día), al igual que en zonas de mucha sombra ya sea provocada por postes de luz, árboles o cables. También se consideraran calles o carreteras de concreto o concreto hidráulico, ya que los resultados pueden variar debido al estriado que se les realiza para el desvío de la lluvia, provocando que se obtengan resultados erróneos.

Se quedan también como trabajo futuro dos aspectos importantes, el primero consiste en generar una base de datos propia que contenga tres clases balanceadas de averías: piel de cocodrilo, bache y grieta, que permitan mejorar las métricas de desempeño de la metodología propuesta. El segundo consiste en el desarrollo de la aplicación móvil (última etapa de la metodología propuesta).

Referencias

1. Arriola-Oregui, I.: Detección de objetos basada en deep learning y aplicada a vehículos autónomos. Master's Thesis, Universidad del País Vasco (2018)
2. Arya, D., Maeda, H., Ghosh, S. K., Toshniwal, D., Sekimoto, Y.: RDD2020: An annotated image dataset for automatic road damage detection using deep learning. Data in Brief, vol. 36, pp. 107133 (2021) doi: 10.1016/j.dib.2021.107133

3. Arya, D., Maeda, H., Sekimoto, Y.: From global challenges to local solutions: A review of cross-country collaborations and winning strategies in road damage detection. Advanced Engineering Informatics, vol. 60, pp. 102388 (2024) doi: 10.1016/j.aei.2024.102388
4. Ashwini, K., Bhagwat, G., Sharma, T., Pagala, P. S.: Trigger-based pothole detection using smartphone and OBD-II. In: Proceedings of the IEEE International Conference on Electronics, Computing and Communication Technologies, vol. 6, pp. 1–6 (2020) doi: 10.1109/conecct50063.2020.9198602
5. Azar, M. A., García, J. L., Bernal, S., Aleman, L., Tolaba, M.: Inteligencia artificial aplicada a IoT. In: XXII Workshop de Investigadores en Ciencias de la Computación, pp. 54–59 (2020)
6. Azar, M. A., Tapia, M., García, J. L., Pérez, A. J. M.: Inteligencia artificial de las cosas. In: XXI Workshop de Investigadores en Ciencias de la Computación (2019)
7. Azar, M. A., Tapia, M. A.: Detección de averías viales mediante IoMT aplicada a smart cities. In: XXIV Congreso Argentino de Ciencias de la Computación, pp. 1133–1141 (2018)
8. Dhiman, A., Klette, R.: Pothole detection using computer vision and learning. IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 8, pp. 3536–3550 (2019)
9. Flórez-Pareja, L. D., Escobar-Arenas, J. P., Fernandez-Mc-Cann, D. S.: Estimación de irregularidades en pavimentos mediante técnicas de procesamiento digital de imágenes. Revista Politécnica, vol. 19, no. 37, pp. 20–28 (2023) doi: 10.33571/rpolitec.v19n37a2
10. Guo, G., Zhang, Z.: Road damage detection algorithm for improved YOLOv5. Scientific Reports, vol. 12, no. 1 (2022) doi: 10.1038/s41598-022-19674-8
11. Instituto Nacional de Estadística y Geografía: Encuesta nacional de seguridad pública urbana (2024) www.inegi.org.mx/programas/ensu/
12. Khokhar, N.: Pothole detection using python and deep learning (2023) github.com/noorkhokhar99/Pothole-Detection-Pothole-Detection-using-python-and-deep-learning
13. Manzanares-González, A.: Detector de baches con deep learning. Master's Thesis, Universitat Pompeu Fabra Barcelona (2019)
14. Ramírez Venegas, C. A.: Sistema computacional para detectar anomalías en vías urbanas con base en vibraciones mecánicas. Master's Thesis, Centro de Investigación en Computación, Instituto Politécnico Nacional (2015)
15. Raschka, S., Mirjalili, V.: Python machine learning: Machine learning and deep learning with python, scikit-learn, and tensorflow 2. Packt Publishing (2017)

ISSN 1870-4069

Conformación dinámica de equipos colaborativos en un sistema multiagente ambiental

Manuel Hernández, Eduardo Sánchez-Soto

Universidad Tecnológica de la Mixteca,
Instituto de Computación,
México

{manuelhg, esanchez}@mixteco.utm.mx

Resumen. En este artículo se plantea la utilización de un formalismo de manejo de equipos en sistemas distribuidos aplicado a un conjunto de componentes autónomos que manejan una historia individual de sus propias acciones y representaciones del ambiente, incluidas las interacciones de los componentes autónomos entre sí. Tal historia (narrativa) es tratada a través de un cálculo de eventos, el cual no sólo registra los eventos que ocurren, sino que además reporta qué propiedades del sistema de agentes (vistos como componentes autónomos) son verdaderas en un instante dado. Como resultado de este trabajo mostramos la posibilidad de conformar equipos de trabajo de agentes para llevar a cabo tareas realizables colectivamente.

Palabras clave: Colaboración, autonomía, ambiente.

Dynamic Formation of Collaborative Teams in an Environmental Multi-agent System

Abstract. This article proposes the use of a team management formalism in distributed systems applied to a set of autonomous components that manage an individual history of their own actions and representations of the environment, including the interactions of the autonomous components with each other. Such a history (narrative) is treated through an event calculus, which not only records the events that occur, but also reports which properties of the agent system (seen as autonomous components) are true at a given moment. As a result of this work, we show the possibility of forming working teams of agents to carry out collectively achievable tasks.

Keywords: Teamwork, autonomy, environment.

1. Introducción

En este trabajo se propone abordar el problema de la colaboración entre agentes (componentes autónomos) a través de la conformación de equipos en un sistema distribuido ambiental.

Se plantea que tal problema puede abordarse mediante un formalismo conocido como SCEL [1] y un cálculo de eventos [8]. Este enfoque proporciona mecanismos computacionales para que algunos agentes autónomos se informen colectivamente de su entorno y, para nuestro caso, ésta información brinde un planteamiento de gestión de equipos de agentes. Así, se permite a los agentes el gestionar equipos basado en la percepción de los cambios en su entorno local y compartirlo con uno o más de los agentes autónomos existente para de esta manera tener mayores posibilidades de acciones colectivas o individuales, y suponiendo que tales acciones que modifican, a su vez, el ambiente en donde existen.

Nuestro planteamiento considera a los agentes (componentes autónomos) dentro de un ambiente sin predefinirles un modelo del mundo, y en su lugar la interacción directa del agente con su entorno genera modelos conforme pasa el tiempo. Aquí, empleamos una versión especializada del formalismo SCEL para aplicarlo a un sistema de agentes robóticos con interfaces, junto con un cálculo de eventos, y de esta manera garantizar una dinámica de formación de equipos y de posibles coaliciones (conjuntos de equipos).

Para obtener decisiones razonables de membresía a uno u otro equipo por parte de los agentes, cada agente puede recopilar, compartir o recibir información de su entorno, generando una noción de verdad consensuada y colectiva que puede aplicarse a la creación de equipos de trabajo como parte del diseño y ejecución de planes o bien como respuesta a labores que requieren atención inmediata. El cálculo de eventos aquí aplicado proporciona un marco computacional para representar y razonar sobre un ambiente dinámico, generando modelos lógicos que cambian con el tiempo. Al combinar éste cálculo con la noción de interfaces de SCEL, podemos controlar la interacción del agente.

Las interfaces permiten al agente percibir y actuar sobre su entorno, proporcionando un esquema de comunicación entre el agente y el mundo exterior [18], así como con otros agentes. De la misma forma, la percepción del ambiente posibilita, entre otras actividades, la gestión de equipos (la importancia de cómo el percibir el ambiente influye en la toma de decisiones grupales está descrito para un caso visual en [12] y en general en [3]).

El cálculo de eventos e interfaces proporcionan mecanismos computacionales adicionales para que los agentes se informen mejor de su entorno [5]. Al representar eventos y sus relaciones causales, los agentes (o en nuestro ejemplos, agentes robóticos) pueden crear equipos, actualizar sus membresías, y ampliar o eliminar tales equipos, para así anticipar cambios y tomar decisiones de acción basadas en un manejo efectivo de colaboración. Veremos que el conocimiento compartido es un aspecto fundamental que permite a los a los agentes alcanzar objetivos específicos para la conformación de equipos.

Panorama de este trabajo. En la Sección 2 se da una introducción al formalismo SCEL de Rocco de Nicola et al. [15], acentuando las características que un agente autónomo debe poseer para apoyar la formación de equipos; se menciona aquí que un concepto a resaltar es el de vector de interfaces. En la Sección 3 se plantea la aplicación especializada de un cálculo de eventos ideado por Robert Kowalski y Marek Sergot [8] como complemento a la decisión de cómo se debe gestionar el tema de equipos colaborativos entre agentes.

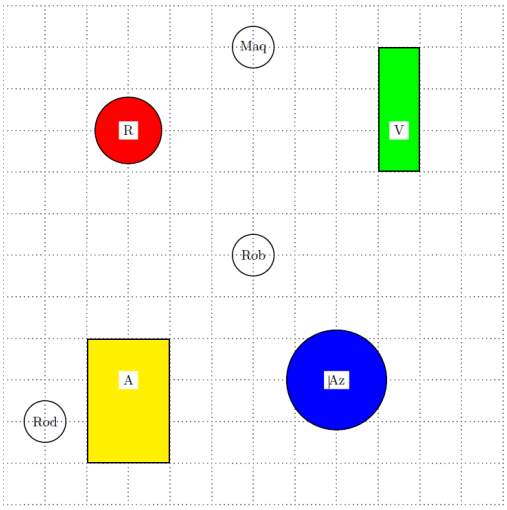


Fig. 1. Escenario inicial (V=verde, A=amarillo, R=rojo, Az=azul).

En la Sección 4 se ejemplifica nuestra propuesta tratando un par de casos de estudio de un conjunto de agentes robóticos explorando un ambiente. Al suponer objetivos a cumplir, y dada la información obtenida del entorno, se puede optar por formar equipos colaborativos para cumplir con sub-objetivos o los objetivos completos. Finalmente, se presentan algunas conclusiones en la sección final.

2. SCEL y la formación dinámica de equipos

Sea un conjunto de agentes autónomos conviviendo entre sí con la posibilidad de agruparse estructuradamente para llevar a cabo tareas grupales. Este tipo de agentes autónomos pueden considerarse como un tipo de robots inmersos en un ambiente [18]. Los agentes tienen posibilidades de comunicación, además de un apoyo informativo externo a cada agente pero interno al sistema (el equivalente a un depósito o nube local de almacenamiento compartido).

Se aprovechará la noción de interfaz de un componente autónomo para una aplicación de un cálculo lógico de eventos (que tiene a su vez una formulación natural en términos de programación lógica [7, 9, 2], con implementaciones en Prolog [10]) además del tratamiento de equipos (proponiendo la creación de un equipo, inscribiéndose a uno ya existente, abandonando uno, o eventualmente, eliminar otro).

Para alcanzar el objetivo planteado, se ha ideado una arquitectura modular en un sistema de agentes, en donde cada componente autónomo o agente (robótico) es un módulo computacional y reactivo (capaz de realizar físicamente acciones) basado en una computadora (Raspberry Pi) ejecutando Linux y otros módulos auxiliares más basados en microcontroladores. Dentro de este conjunto de módulos auxiliares basados en microcontroladores, se tienen 3 tipos: de recabado de datos sensoriales, de control de actuadores, y de gestores de comunicación Wi-Fi (por ejemplo, como puntos de acceso).

La técnica de comunicación mediante sockets y TCPs ha sido fundamental, ya que permiten la apertura de canales de comunicación con mensajes económicos y en tiempo real. Debido a estas posibilidades de comunicación, existe una dinámica de equipos: sea que un plan puesto en marcha considere el formar equipos, o sea que los equipos posibiliten el diseño de ciertos planes. En [16] se propone una formulación pragmática de tal formación de equipos basada en emitir o recibir notificaciones para organizar posible trabajo colaborativo. En este trabajo mostramos como las abstracciones tomadas del formalismo SCEL son lo suficientemente generales para apoyar también a la programación declarativa (representada en este caso por la programación lógica).

Los autores de [15] señalan una posibilidad de su formalismo: las interfaces contienen predicados que cambian con el tiempo. Esto nos lleva a ver que SCEL es adaptable también a una formulación particular y práctica de un cálculo lógico de eventos [8, 14], en donde los predicados que cambian con el tiempo son llamados fluentes; estos fluentes se utilizarían en la toma de decisiones de los componentes autónomos desde el punto de vista de narrativas peculiares o individuales (en contraste a narrativas generales), así mostrando que, con adecuadas restricciones, este cálculo de eventos tiene potencial para brindar racionalidad temporal e histórica en la toma de decisiones acerca de la gestión de equipos de agentes.

2.1. Descripción sucinta de un formalismo para el estudio de componentes autónomos: SCEL

Según SCEL, un componente autónomo (CA) dentro de un sistema distribuido se caracteriza por sus atributos e interfaces. (Tales componentes autónomos pueden ser vistos como agentes autónomos en el contexto de un agente que interactúa con su ambiente [18].) En la parte correspondiente a los atributos, se incluyen descripciones de los CAs tales como identidad, capacidades, coordenadas espaciales (ubicación), membresías a grupos, niveles de confianza, tiempos de respuesta, entre otros.

Particularmente, dentro de estos atributos y con miras a posibles integraciones grupales, cada CA consta de una o varias etiquetas que identifican su disponibilidad (o la ausencia de) para aceptar una membresía a un grupo de CAs a través de interfaces. Un CA puede pertenecer a varios equipos o a ninguno, y los mismos equipos pueden formar otros mayores llamados coaliciones. Esta conformación de super-equipos se detiene dependiendo de las necesidades de cada aplicación. Los CAs están capacitados para utilizar depósitos de información (DI).

Estos depósitos permiten compartir información que se obtiene de las actividades sensoriales de los agentes (o incluso información obtenida por deducción) para transmitirla previamente procesada o directamente a otros CAs. La ventaja de éste enfoque es el delegar labores a componentes especializados y complementar las fuentes de información sensorial o de deductivamente obtenida, con lo que podría equiparse con información detallada e indirecta a los demás CAs.

En nuestro enfoque, y por que el formalismo SCEL así lo permite (y lo promueve), las conductas de los CAs tienen una dualidad entre planes y secuencias de acciones, y a su vez entre planes y tratamientos de equipos. Estas conductas, entonces, son el resultado o de reaccionar a un ambiente o de racionalizar un estado (o estados del CA) para llegar a generar planes y acciones consecuentes.

El conocimiento que prevalece en una modelación de tipo SCEL tiene dos posibles fuentes: Cuando un CA adquiere tal conocimiento (información) directamente o bien cuando obtiene o complementa su información vía directa o vía depósitos existentes, para que un CA se informe de a cuáles equipos, si es el caso, el CA puede integrarse. Varias modalidades de interacción surgen entonces en la interacción con los depósitos: se puede contribuir con información propia, o bien se puede obtener información; o aún más, puede llegarse al caso de modificar (con los permisos apropiados) la información ya existente en un depósito (modificándola al grado de borrarla).

Las interfaces permiten que los agentes auto-examinen sus propios predicados "cambiantes" o fluentes de su propia situación o puedan adquirir información de otros agentes a los cuales tienen posibilidades de acceder y obtener su información. Con los fluentes, cada agente puede decidir qué es verdad en un tiempo t de su ambiente circundante. La información antes mencionada se ve plasmada a través de un vector de interfaces que consta de n entradas:

in_1	in_2	$\ldots$	in_k	out_1	out_2	$\ldots$	out_l	$pred_1$	$pred_2$	$\ldots$	$pred_m$

(1)

donde: $n = k + l + m$. Las entradas señaladas con *in* se utilizan para adquirir información del ambiente, sea de sensores internos, de servidores de sensores, e incluso de otros CAs. Las entradas señaladas con *out* son para emitir información ya sea de los sensores internos o de aquella otra información adquirida directa o indirectamente (por ejemplo, por medio de otros CAs). Por último, las entradas señaladas con *pred* sirven como aserciones temporales.

Estas aserciones son fórmulas lógicas que cambian con el tiempo, es decir, en la terminología del cálculo de eventos, fluentes. Quienes pueden modificar el valor de estos fluentes son o el mismo CA u otros componentes autónomos (teniendo permisos vía una política en vigor) o bien pueden ser deducidos del cálculo de eventos. SCEL permite restringir a quien se comparte la información de estas casillas, dependiendo la aplicación. Más información y definiciones formales extensas se encuentran en [15].

3. Un cálculo de eventos aplicado a agentes autónomos

Vemos ahora una sucinta descripción del cálculo de eventos. Mayor información puede ser obtenida de [17]. El cálculo de eventos tiene una formulación técnica basada en programas normales de la programación lógica. Algunos predicados del cálculo de eventos son fundamentales:

1. **Initiates** $(\alpha,\ \beta,\ \tau)$: El fluente β comienza a ser verdadero cuando acontece la acción α en el instante τ.

2. **Terminates** $(\alpha,\ \beta,\ \tau)$: El fluente β cesa de ser verdadero cuando acontece la acción α en el instante τ.

3. **InitiallyP** (β) El fluente β es verdadero desde el instante 0.

4. **Happens** $(\alpha,\ \tau)$ La acción α ocurre en el instante τ.

Tabla 1. Objetos seleccionados para nuestro caso de estudio.

Objeto	Características
Objeto 1 (Ob1):	Ciírculo azul (Ca)
Objeto 2 (Ob2):	Rectángulo amarillo (Ra)
Objeto 3 (Ob3):	Círculo rojo (Cr)
Objeto 4 (Ob4):	Rectángulo verde (Cv)

5. **HoldsAt** $(\beta,\ \tau)$ El fluente β es verdadero en el instante τ.
6. **Clipped** $(\tau_1,\ \beta,\ \tau_2)$: El fluente β cesa de ser verdadero en el intervalo $(\tau_1, \tau_2]$.
7. **Releases** $(\alpha,\ \beta,\ \tau_2)$: El fluente β no es ya más "inercial" después del evento α ocurrido en el instanteτ.
8. **InitiallyN** (β): El fluente β es supuesto como falso desde el instante 0.
9. **Declipped** $(\tau_1,\ \beta,\ \tau_2)$: El fluente β es verdadero desde algún instante que está en el intervalo $(\tau_1,\ \tau_2)$.

Damos a continuación una formulación de índole intuitiva que abarca los conceptos que utilizaremos posteriormente. Primero, se tiene una ley inercial tipo 0: Si suponemos que en un tiempo 0 la aserción A es verdadera (**HoldsAt**$(A,\ t)$), y dado un $t > 0$, la aserción A sigue siendo verdadera, a menos que un evento, que haya ocurrido entre 0 y t, haya cambiado su valor. Una generalización es una ley inercial tipo 1: Ahora un evento e ocurre en un tiempo t_1 (**Happens**(E, t_1)), que hace que A sea verdadera.

Para un tiempo t_2, con $t_2 > t_1$ A sigue siendo verdadera a menos que un evento e1 que ocurra entre t_1 y t_2 haga a A falsa. Finalmente, A es truncable (**Clipped**$(t_1,\ A,\ t_2)$) entre t_1 y t_2 si existe un evento entre t_1 y t_2 tal que haga a A falsa entre t_1 y t_2. En otras palabras, para una ley inercial 0, con $t_1 > 0$, A es verdadero en t_1 si A no fue truncable entre 0 y t_1. Para una ley inercial tipo 1, A no es truncable entre t_1 y t_2, con $t_1 < t_2$, y t_1 siendo el instante en que A comienza a ser verdadero.

Para un problema como el de agentes autónomos como el que estamos tratando, la fundamentación axiomática del cálculo de eventos es la siguiente: Cada CA ignora todo acerca de su ambiente en un tiempo inicial, pero puede tener información inicial propia por auto-inspección. Cada evento de un CA al adquirir información sensorial de un objeto O del ambiente hace que su información involucre a O. Cada vez que un CA se comunica con otro CA el acervo de información de ambos se comparte, haciendo que cada CA tenga mayor o igual o información a la previamente ya adquirida. El fluente de "contactar" un agente a otro es modificado después de que dos agentes se encuentran.

Notemos, entonces, que un CA siempre adquiere información incrementalmente. Notemos también que la información ya poseída tiene que compatibilizarse con la recibida. Cada CA, además, tiene un manejo de tiempo que es global. Es fundamental cerciorarse que éste cálculo de eventos es implementable según la programación lógica, pero reconociendo que el tipo de historia que cada CA maneje puede ocasionar problemas en la saturación de memoria, lo que obliga a tomar posibles estrategias para el almacenamiento de datos (como el borrar datos antiguos o bien algunos marcados

como de baja prioridad). De todas formas, algunos detalles pragmáticos representan de por sí sus propios desafíos. Se mencionan ahora algunas condiciones de aplicabilidad del cálculo de eventos, para lo cual es necesario identificar los siguientes conjuntos:

1. Un conjunto de acciones o eventos (ambos identificados como el mismo conjunto).
2. Un conjunto de fluentes, que son predicados con valor cambiante dependiendo de las acciones que van ocurriendo; cada fluente debe estar relacionado con al menos un evento, sea para que el fluente comience a ser verdadero por la ocurrencia del evento o el fluente comience a ser falso por tal ocurrencia (la liberación de los fluentes de la influencia de eventos es posible, para ciertas aplicaciones, pero tiene que señalarse explícitamente).
3. Un conjunto de valores de tiempo, los cuales pueden ser momentos o intervalos; tales valores deben ser comparables, discretizables (hasta un grado necesario), y de tal forma que toda acción tenga un momento de ocurrencia y todo fluente tenga un valor definido de veracidad dado un momento en el que el fluente se inspeccione. Hablamos de intervalos si tenemos dos momentos t_1 y t_2, con $t_1 < t_2$ tal que el conjunto $\{t,\ t_1 < t \text{ y } t < t_2\}$ sea no vacío.

También es necesario considerar restricciones de integridad, como aquellas suposiciones, leyes, o condiciones de la realidad que son explícitamente establecidas y que los eventos junto con los fluentes deben cumplir en cada ocasión (de otra manera, estarían ocurriendo inconsistencias o violaciones de temporalidad, por ejemplo) y una teoría de causalidad, que condiciona de forma causal a los eventos y a los fluentes. Para obtener una aplicación adecuada del cálculo de eventos, aquí se listan algunos temas que se deben abordar dependiendo la aplicación en mente:

Tipos de narrativas. Una narrativa es la identificación de un conjunto de ocurrencias de eventos en el tiempo. Las narrativas aquí elaboradas son individuales, por cada agente existente. Notaremos que el compartir tales narrativas permite una diseminación de información para una mejor toma de decisiones.

Granularidad. Se supondrá una descripción del tiempo discretizada, con respecto a un grado de granularidad, lo suficiente para no perder los instantes de la ocurrencia significativa de eventos que modifiquen fluentes. También manejaremos una granularidad espacial, para que la movilidad de los robots sea descrita por coordenadas enteras [13].

Intervalos. Todos los intervalos serán supuestos sobre el tiempo discretizado y serán abiertos por la izquierda y cerrados por la derecha; tales intervalos $(t_1, t_2]$ se describen como conjuntos discretizados con elementos t cumpliendo que $t > t_1$ y $t \leq t_2$.

4. Escenario de agentes robóticos como componentes autónomos

Seguimos algunas ideas de SCEL para la configuración de un escenario robótico (nombrando robot a un agente autónomo, en adelante) en donde nombraremos a los robots Rob, Maq y Rod como tres robots (que jugarían también el papel de componentes autónomos, en la terminología de SCEL) que exploran en un ambiente relativamente

controlado. Se supondrá que los agentes robóticos están conectados a una red local de alta confiabilidad, de forma inalámbrica mediante tecnología Wi-Fi. Notemos que esto conlleva una arquitectura de equipos de trabajo [6], cuando se considera la automatización de las interacciones entre los componentes autónomos. Como supuestos de movilidad (ver Figura 1) tenemos lo siguiente: Cada robot puede moverse a lo largo y ancho de la cuadrícula señalada en esta figura, en la dirección hacia adelante (f), hacia atrás (b), a la derecha (r) o a la izquierda (l), en unidades enteras (siendo la actual cuadrícula una de esquina inferior izquierda ubicada en $(-6,\ -6)$ y en esquina superior derecha ubicada en $(6,\ 6)$). Consideremos un escenario como el de la Figura 1.

En este escenario existen los tres agentes robóticos mencionados, Rod, Rob y Maq, que bien pueden ser de naturaleza heterogénea, pero es fundamental que mantengan un conjunto uniforme de interfaces. En la Figura 1 las figuras sombreadas son obstáculos sólidos, detectables por cierto tipo de sensores (ultrasónicos, por ejemplo), e impenetrables e indeformables por los robots. Bajo la instancia de encuentro casual entre dos robots (encuentro considerado como un evento), los fluentes de cada robot cambian de "información no compartida" a "información compartida", de tal forma que los robots se comunican entre sí y se mandan la información de qué lugares están ocupados por objetos y qué objetos son. Planteamos los siguientes a antecedentes:

1. Hay tres robots en este mundo. Todos tienen una identidad.
2. Hay objetos que son circulares: C1, C2, ...
3. Hay objetos rectángulares, R1, R2, ...
4. Cada objeto tiene asociado un color (rojo, amarillo, verde o azul) y una ubicación (tomada como el centroide del objeto).

Por ejemplo, podríamos considerar una interfaz (tipo vector) que conste de un par de celdas del vector de interfaz siendo celdas recepción de información, in_1 e in_2; otro par de celdas del vector siendo celdas de emisión de información out_1 y out_2; y otras algunas entradas relacionadas con predicados que pueden cambiar, tales como el o los equipos a los que pertenece el robot ($pred_1$), el nivel de energía del robot ($pred_2$), su ubicación ($pred_3$), así como la información de los objetos que vayan adquiriendo conforme exploran el mundo en donde se hayan inmersos ($pred_4$). El vector de interfaz para estos robots quedaría así:

in_1	in_2	out_1	out_2	$pred_1$	$pred_2$	$pred_3$	$pred_4$

(2)

El fluente $pred_4$ se puede desglosar así (pues la celda en sí puede ser lo suficientemente compleja como fórmula lógica, incluidos los sujetos o estructuras de datos): un fluente es K, que consta de una lista de objetos conocidos por cuenta propia, de conocimiento "experimentado", K-DB, del tipo:

```
conozco1(objeto, características, ubicación, instante).
```

Y otra más, KA-DB, de objetos conocidos por comunicación con otros robos o de conocimiento "adquirido", de tipo:

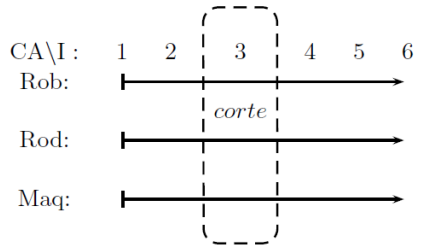


Fig. 2. Líneas de tiempo para los agentes robóticos Rob, Rod y Maq. Aquí, CA=componente autónomo, e I=instante.

```
conozco2(objeto, características, ubicación, instante).
```

Sea que los robots exploran su mundo, el cual contiene algunos objetos geométricos coloreados dispersos (ver la descripción en la Figura 1). La exploración de los robots se lleva a cabo en una área delimitada. Al estar dentro de una zona cercana a un objeto, lo reconocen, vía algún mecanismo sensorial (hemos utilizado robots con cámaras que reconocen códigos QR, por ejemplo). Del conocimiento propio experimentado K por un agente, no hay ninguna duda de su veracidad desde el punto de vista del propio agente.

Del conocimiento adquirido, KA, se pueden idear mecanismos de consenso grupal para así fortalecer la salvaguarda del sistema contra información falsa o maliciosa. Vamos a suponer que algunas entradas de la interfaz vectorial como la K-DB o la KA-DB. éstos fluentes contribuyen a la toma de decisiones con respecto a la creación, de equipos de trabajo que realizarían tareas grupales, así como a la modificación de una membresía, la modificación o eliminación de tales equipos.

Por ejemplo, se puede considerar como un atributo el número de objetos conocidos, el número de objetos con cierta forma geométrica o color, o bien el número total de objetos conocidos, que bien puede obtenerse de la suma de objetos conocidos por experiencia propia y de aquellos conocidos por conocimiento adquirido ajeno. Tomamos éste último criterio, para ejemplificar.

Debido a que no se conoce de antemano por los agentes el número total de objetos, no se tendría una condición de finalización en el tiempo, sino solo posibles "cortes" de información compartida en instantes determinados. En el diagrama de la Figura 2 (inspirado de técnicas descriptivas de cómputo y sistemas distribuidos [11]), se grafican alguas líneas de tiempo. Se muestra un corte (o instantánea) en el instante 3 encerrado en un rectángulo hecho de segmentos entrecortados.

Cada línea representa el tiempo de existencia de cada agente. Con mayores decoraciones gráficas sobre cada línea de tiempo, como veremos, se pueden representar eventos que acontecen a los agentes durante el transcurso del tiempo. Dos eventos nos interesan en particular: el evento en el que un agente conoce un objeto y las características de este objeto, y otro evento que consiste en que simultánemente un evento de envío de mensajes acontece con su recepción[1], así logrando que el conocimiento entre agentes sea diseminado.

[1]Esta simultaneidad no sería realista en general en sistemas distribuidos, por el retardo de tiempo entre el envío y recepción del mensaje, pero que que aquí abstraemos tal situación.

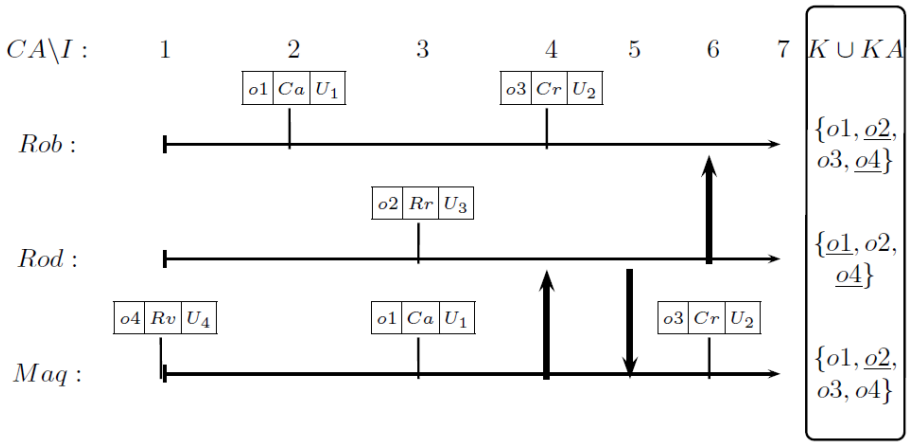


Fig. 3. Diagrama de temporalidad (timing diagram), con instantes cuando cada robot (agente) obtiene información de su entorno. En el rectángulo de la derecha está la información recopilada por cada agente hasta el instante 7.

El primer evento es representado por tres casillas anexas mediante un segmento a una línea de tiempo. El segundo evento es representado por un segmento grueso y dirigido (señalando el remitente y el destinatario (receptor) de un mensaje informativo acerca de los objetos conocidos) que conecta, ortogonalmente, dos líneas de tiempo. En la Figura 3 se muestra el uso de este tipo de diagramas para representar una historia de tiempo, que consta de una secuencia ordenada de instantáneas o (del inglés y de la terminología de sistemas distribuidos, snapshots) o cortes. En este caso particular, cada U_i es la ubicación de un objeto, que bien podría ser información relevante como un criterio a seguir para conformar equipos en otros contextos.

Consideremos una narrativa asociada al agente Rob, desde su punto de vista: En el instante 2 adquirí conocimiento acerca de la existencia del objeto 1, el cual es un círculo de color azul. En el instante 4 aquirí conocimiento del objeto 3, el cual es otro círculo de color rojo. En el instante 6 recibí informacion del agente Rod, quien me hizo conocer que existe un objeto 1, circular, de color azul (redundante, pero comprueba consistencia de informacion); tambien en ese mismo paquete de informacion se me informo del objeto 4, rectangular, de color verde, y del objeto 2, rectangular, de color rojo. Para el instante 7 conozco los objetos $\{o1,\ o2,\ o3,\ o4\}$ asi como sus formas y colores asociados. Este es un ejemplo de eventos identificables en esta narrativa:

a) AdquirirConocimiento/2,

b) EnviarInfo/2,

c) RecibirInfo/2 (aquí el número adjunto representa la aridad del predicado).

A su vez, algunos fluentes identificables serían:

a) NotificadoAgente(A,B): B es notificado de alguna información por A;

b) Conoce(A,Ob): el agente A conoce información (detallada) acerca del objeto Ob.

Habiendo ya tratado un ejemplo de colaboración general con el objetivo de recabar información de un ambiente ahora presentamos otro ejemplo, pero esta vez involucrando otro agente (robótico) llamado Ben. Sea que acontecen los siguientes eventos, en donde por cada instante de tiempo t se anexa de forma sufija su tiempo de ocurrencia (notemos que esto nos permite describir eventos simultáneos), consecutivamente:

- **Happens**(Hallar(Maq, Rv), t1).
- **Happens**(Hallar(Rob, Ca), t2).
- **Happens**(Hallar(Ben, Rr), t2).
- **Happens**(Hallar(Rod, Rr), t3).
- **Happens**(Hallar(Maq, Ca), t3).
- **Happens**(ComunicaHallazgos(Rob,Ben),t4).
- **Happens**(ComunicaHallazgos(Maq,Rod),t4).
- **Happens**(ComunicaHallazgos(Rod,Maq),t5).
- **Happens**(ComunicaHallazgos(Rod,Rob),t6).

Ahora, dos equipos pueden formarse, dependiendo de si se tiene información de la existencia de ciertos objetos. Por un lado, a partir del instante 7 los agentes Maq, Ben y Rob son quienes tienen información acerca de los objetos $\{o1,\ o2,\ o3\}$. Por otro lado, a partir también del instante 7 se tiene información de que los agentes Rob, Rod y Maq conocen los objetos $\{o1,\ o2,\ o4\}$:

- **Happens**(ConformanEquipo(Equipo1([Maq, Ben, Rob])),s).
- **Happens**(conformanEquipo(Equipo2([Rob, Rod, Maq])),u).

Notemos que aquí el ejemplo supone un corte hecho en el instante 7, para inspeccionar en eso momento qué agentes conocen qué. Bajo cortes previos, otros equipos podrían haberse también conformado, o bien, si el tiempo sigue fluyendo, es posible que algunos miembros abandonen sus equipos (bajo algún criterio) o bien algunos equipos desaparezcan como tales.

Aunque el corte lo hemos realizado a partir del instante 7, notemos que el criterio de información puede servirnos para formar equipos en instantes previos. Por ejemplo, como es mostrado en el diagrama 4, para el instante t_3 se sabe que Rod y Ben conocen el objeto o_2, y esto puede ya servir de guía para formar un equipo. Similarmente, para el instante t_4, Maq y Rob conocen el objeto o_1, y este también puede ser de guía para formar otro equipo.

Dada una propiedad de pertenencia (como parte de un vector de interacción), para tiempos $s > 7$ y $u > 7$ (suponiendo una ley inercial tipo 1) Equipo 1 y Equipo 2 son equipos existentes, y cualquiera de éstos equipos podrán realizar tareas colaborativa que involucren los objetos conocidos. Notemos que un fluente de pertenencia a un equipo es aplicable a un agente y equipo dados, y puede cambiar de falso a verdadero o de verdadero a falso con el tiempo.

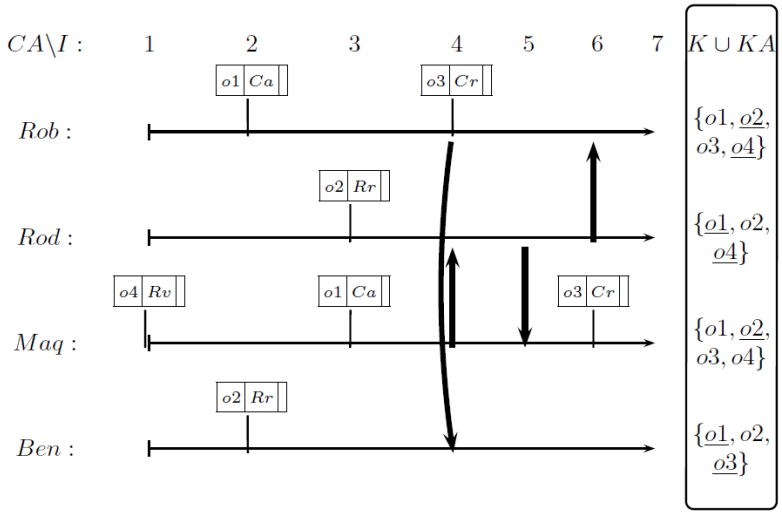


Fig. 4. Diagrama de temporalidad con un agente agregado, ilustrando el punto de conformación de equipos de acuerdo con el conocimiento adquirido por cada agente a través del tiempo. En el rectángulo a la derecha mostramos qué información tiene cada agente en el instante 7.

Ahora, para un tiempo t superior al instante 7, y de seguirse un criterio simple de intersección sobre los conjuntos de objectos conocidos, tendríamos:

– **HoldsAt**(Equipo1([Maq,Ben,Rob]),t).

Lo que implica que Maq, Ben y Rob son quienes conocen $\{o_1, o_2, o_3\}$, y:

– **HoldsAt**(Equipo2([Rod,Rob,Maq]),t).

Lo que implica que Rod, Rob y Maq son quienes conocen $\{o_1, o_2, o_4\}$. A partir de este instante t, los agentes Maq, Rod, Rob y Ben pueden, como ejemplos de decisiones y acciones:

a) Rechazar membresías, bajo algunas condiciones.

b) Optar por salir de un equipo (notemos que sería motivo de incorrección el salir de un equipo sin estar como miembro).

c) Conformar coaliciones, al conjuntar dos o más equipos.

d) Heredar las condiciones bajo las cuales los equipos son útiles (por ejemplo, si una tarea que requería un equipo ya fue hecha, es posible que el equipo carezca de sentido su existir).

Otras narrativas individuales pueden obtenerse de forma similar del diagrama de la Figura 4. De la unión de narrativas individuales surgirían narrativas grupales, mismas que pueden dar indicios más generales que las individuales para analizar el sistema, y así para el caso posible de una narrativa global.

Cabe destacar que una forma de seguridad (o de confiabilidad) del desarrollo (runtime) del sistema sería el cotejar (por algunos agentes, o todos) conocimientos acerca de los objetos y sus atributos, así como posibles marcas de tiempo (timestamps) para mostrar la consistencia de la información obtenida durante el tiempo transcurrido hasta un instante dado t_m, con $m > 1$. Otra índole de información tal como la ubicación de los objetos sería útil también, dependiendo la aplicación.

En la Figura 3, en la columna indicada con K, se ha recabado el total de objetos conocidos a partir del instante 7; se han señalado en los conjuntos de objetos conocidos aquellos que fueron conocidos por experiencia propia y aquellos otros conocidos por comunicación (indicados por nombres de objetos subrayados). Supongamos que HallazgoObjetos es un atributo que es visible desde las interfaces de los componentes autónomos del sistema. Para el instante 7, como hemos visto, es posible formar equipos que están determinados por aquellos agentes que conocen algunos objetos, asi logrando que el cálculo de eventos ayude en la toma de decisiones para conformar equipos [4, 15].

5. Conclusiones

Se ha afirmado en este trabajo que SCEL sí brinda un apoyo teórico a la formación de equipos y en principio, a coaliciones (conjuntos de equipos). Además, también se ha mostrado que a través de la noción de temporalidad se puede coadyuvar a la dinámica de equipos de agentes. A través de la definición de interfaces, SCEL propone un mecanismo de formación de equipos conforme a las necesidades del sistema que se presentan. La literatura indica la importancia de una adecuada comunicación entre agentes [19] para una acertada gestión de equipos.

Para una formulación de una jerarquización (no existente en nuestra propuesta) que sea útil en el consenso de datos, ver [13]. En la parte colaborativa, sería posible tratar el tema de acuerdo con la teoría de juegos, brindando recompensas a los agentes más proactivos y participativos, estableciendo esquemas de negociación, y en la parte competitiva o de adversarios, se esperaría integrar equipos defensivos o de ataque.

Referencias

1. De-Nicola, R., Latella, D., Lafuente, A. L., Loreti, M., Margheri, A., Massink, M., Morichetta, A., Pugliese, R., Tiezzi, F., Vandin, A.: The SCEL language: Design, implementation, verification (2015) doi: 10.1007/978-3-319-16310-9_1
2. Doets, K.: From logic to logic programming. The MIT Press (1994)
3. Dunin-Keplicz, B., Verbrugge, R.: Awareness as a vital ingredient of teamwork. In: Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 1017–1024 (2006) doi: 10.1145/1160633.1160815
4. Dunin-Keplicz, B., Verbrugge, R.: Teamwork in multi-agent systems: A formal approach. John Wiley and Sons (2010)
5. Hernández-Gutiérrez, M., Sánchez-Soto, E.: Actividad de agentes robóticos regulada a través de información de temporalidad vía un cálculo lógico de eventos. Research in Computing Science, vol. 6, no. 152, pp. 7–20 (2023)
6. Kaminka, G. A., Frenkel, I.: Towards flexible teamwork in behavior-based robots. In: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 1355–1356 (2005) doi: 10.1145/1082473.1082770

7. Kowalski, R.: Logic for Problem Solving. North–Holland (1979)
8. Kowalski, R., Sergot, M.: A logic-based calculus of events. New Generation Computing, vol. 4, pp. 67–95 (1985) doi: 10.1007/BF03037383
9. Krzysztof, R. A.: Logic programming. Formal methods and semantics, Elsevier, vol. B, chapter 10, pp. 493–574 (1990)
10. Krzysztof, R. A.: From Logic Programming to Prolog. Prentice Hall (1997)
11. Kshemkalyani, A. D., Singhal, M.: Distributed computing: Principles, algorithms, and systems. Cambridge University Press (2008)
12. Kulyk, O., van-der-Veer, G., van-Dijk, B.: Situational awareness support to enhance teamwork in collaborative environments. In: Proceedings of the 15th European Conference on Cognitive Ergonomics: The Ergonomics of Cool Interaction, Association for Computing Machinery, pp. 1–5 (2008) doi: 10.1145/1473018.1473025 https://doi.org/10.1145/1473018.1473025
13. Luotsinen, L. J., Bölöni, L.: Role-based teamwork activity recognition in observations of embodied agent actions. In: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems, vol. 2, pp. 567–574 (2008)
14. Mueller, E. T.: Chapter 17: Event calculus. Foundations of Artificial Intelligence, vol. 3, pp. 671–708 (2008)
15. Nicola, R. D., Loreti, M., Pugliese, R., Tiezzi, F.: A formal approach to autonomic systems programming: The SCEL language. ACM transactions on Autonomous and Adaptive Systems, vol. 9, no. 2, pp. 1–29 (2014) doi: 10.1145/2619998
16. ROS: ROS - robotic operating system (2024) www.ros.org/
17. Shanahan, M.: The event calculus explained. Artificial Intelligence Today: Recent Trends and Developments, pp. 409–430 (1999) doi: 10.1007/3-540-48317-9_17
18. Weyns, D., Schumacher, M., Ricci, A., Viroli, M., Holvoet, T.: Environments in multiagent systems. The Knowledge Engineering Review, vol. 20, no. 2, pp. 127–141 (2005) doi: 10.1017/S0269888905000457
19. Zhang, Y., Volz, R. A., Loerger, T. R., Yen, J.: A decision-theoretic approach for designing proactive communication in multi-agent teamwork. In: Proceedings of the ACM Symposium on Applied Computing, Association for Computing Machinery, pp. 64–71 (2004) doi: 10.1145/967900.967917

ISSN 1870-4069

Análisis de impacto en clasificadores CNNs ante la evaluacion de imagenes con perturbaciones naturales

Robin A. Rojas-Alvarez[1], Ivan Reyes-Amezcua[2],
Andres Mendez-Vazquez[2]

[1] Universidad de Guadalajara,
Ingeniería en Nanotecnología,
Mexico

[2] Centro de Investigación y de Estudios Avanzados,
Departamento de Computación,
México

{ivan.reyes, andres.mendez}@cinvestav.mx,
robin.rojas@alumnos.udg.mx

Resumen. El desarrollo de algoritmos capaces de robustecer las redes neuronales profundas se ha convertido en parte esencial de la metodología en la creación de soluciones actuales. Sin embargo, estando en la era del crecimiento y comprensión de las capacidades de las IA, se ha promovido la funcionalidad sobre la seguridad. En este trabajo, se expone el uso de redes neuronales preentrenadas para observar su robustez con el algoritmo de entrenamiento de conjuntos de datos basados en CIFAR-C, identificando puntos clave en la arquitectura para mejorar la respuesta y el comportamiento de las DNNs ante perturbaciones naturales.

Palabras clave: Robustecimiento de algoritmos, CIFAR-C, Fine-tunning, CNN, DNN, preentrenamiento, corrupciones naturales.

Impact Analysis on CNN Classifiers in the Evaluation of Naturally Disturbed Images

Abstract. The development of algorithms capable for robust deep neural networks has become an essential part of the methodology in the creation of current solutions. However, being in the era of growth and understanding the AI capabilities, functionality has been promoted over security. In this paper, we expose the use of pre-trained neural networks to observe their robustness with the CIFAR-C based on dataset training algorithm, identifying key points in the architecture to improve the response and behavior of DNNs to natural disturbances.

Keywords: Algorithm robustness, CIFAR-C, Fine-tunning, CNN, DNN, preentrain, natural corruptions.

1. Introducción

El crecimiento exponencial de las redes neuronales profundas (DNNs, por sus siglas en inglés) ha generado grandes soluciones, destacando en el rubro de las tareas hechas por visión artificial, a su vez se han encontrado nuevas formas de perturbar los algoritmos con alteraciones naturales. En análisis iniciales de casos en DNNs [13] han demostrado que perturbaciones, muchas veces imperceptibles ante el ojo humano, de las imágenes provocaban un cambio en la predicción de los modelos, que pasaba de ser correcta a incorrecta. Los trabajos posteriores de han demostrado la susceptibilidad de las DNNs a las corrupciones naturales no adversarias.

Se han observado diferencias significativas en el rendimiento de las DNNs entre las evaluaciones en condiciones de imagen limpias y las degradadas por perturbaciones naturales (a menudo hasta un 30-40 % de disminución de la precisión) [2], siendo estos resultados motivo de preocupación sobre la fiabilidad de las DNNs a medida que se integran en sistemas con riesgos sociales y de seguridad cada vez mayores. La gran vulnerabilidad de los modelos a los daños infrecuentes y naturales de las imágenes sugiere la necesidad de volver a dar prioridad a nuestra comprensión del rendimiento de los modelos con datos y perturbaciones naturales antes de centrarnos en la resistencia a escenarios de ataques de adversarios, robusteciendo así los algoritmos a estas perturbaciones iniciales.

2. Antecedentes y definiciones

2.1. Redes neuronales profundas y robustecimiento

Las redes neuronales profundas son una composición de funciones y capas computacionales que mapean desde el espacio de entrada en el dominio de la imagen a una predicción. Estas redes están muy parametrizadas, por lo que requieren grandes conjuntos de datos y/o tratamiento de los mismos, en combinación con la optimización de los parámetros (a menudo descenso de gradiente estocástico)[2]. Este artículo se enfocara en las DNNs con aplicaciones en visión artificial.

La robustez en general es un término que se ah adoptado a una serie de interpretaciones en la en la comunidad de la visión por ordenador, incluyendo, entre otras, el rendimiento bruto de la tarea en conjuntos de pruebas, el mantenimiento del rendimiento de la tarea en entradas manipuladas/modificadas, la generalización entre dominios y la resistencia a ataques de adversarios. Sin embargo, todas estas pueden ser características deseadas de la robustez, ya que existen diferentes enfoques, dependiendo el grado de optimización y el tipo de ataque resistente, tomando esto en cuenta, se generalizará a través de la resistencia a alteraciones naturales de los conjuntos de datos que alimentan al algoritmo.

Pruebas recientes sugieren que la robustez de las DNNs se beneficia enormemente de conjuntos de datos a gran escala, ya que contienen más variaciones que podrían ocurrir en el mundo real, llenando la brecha de distribución entre los datos de entrenamiento y los de prueba. Sin embargo, el coste de obtener un conjunto de datos grande y bien definido puede ser excesivamente alto. Por ello, los investigadores generan sintéticamente datos con diferentes aumentos para aumentar la variedad de

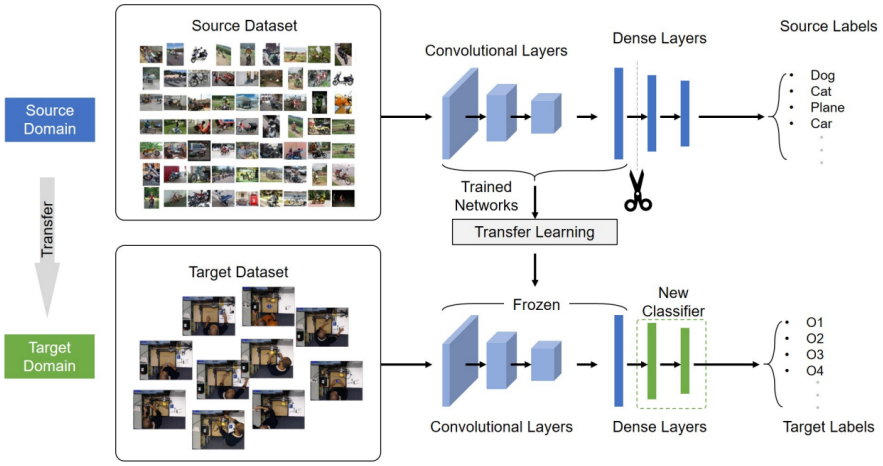


Fig. 1. Arquitectura del modelo de tranfer learning, imágen tomada de [14].

datos y disminuir la brecha de distribución entre los datos de entrenamiento y los de prueba. entre los datos de entrenamiento y los de prueba [16, 14] Mientras que las perturbaciones naturales abarcan desde simples transformaciones como voltear las imágenes, recortar, rotar, trasladar, alteración del color, mejora de bordes, ruido gaussiano hasta redes neuronales generativas adversarias (GANs)

2.2. Fine-tuning y transfer learning

El meta aprendizaje, o aprender a aprender, utiliza los conocimientos previos de una de tareas, algoritmos y evaluaciones de modelos con el fin de obtener mejores resultados, más rápidos y más eficientes cuando se aplican a datos que no se han visto antes. a datos que no se han visto antes [3]. Un problema común del meta aprendizaje es la recomendación de algoritmos, donde dado un conjunto de instancias de problemas P de una distribución D, un conjunto A de algoritmos y una medida de rendimiento m: P x $A \rightarrow \mathbb{R}$, el problema de recomendación de algoritmos consiste en encontrar un mapeo $f : P \rightarrow A$ que optimiza la medida de rendimiento esperada m para instancias P con una distribución D [11]. Se ofrece una definición formal del aprendizaje por transferencia en términos de dominios y tareas de aprendizaje.

Dado un dominio de origen D_s un dominio de destino D_t, y unas tareas de aprendizaje T_s y T_t, el aprendizaje por transferencia pretende mejorar el rendimiento de T_t con conocimientos obtenidos de un dominio D_s y una tarea de aprendizaje T_s diferentes pero relacionados, donde $D_s \neq D_t$, o $T_s \neq T_t$. [9, 10]. En cambio que el ajuste fino de un modelo preentrenado es un enfoque común para llevar a cabo el aprendizaje profundo a medida que los modelos de base están disponibles. Si bien este enfoque mejora el aprendizaje supervisado en varios casos, también se ha observado un exceso de ajuste durante el ajuste fino supervisado en varios casos, también se ha observado sobreajuste durante el ajuste fino, Figura 1 .

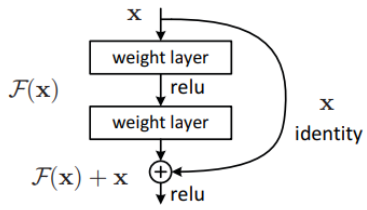


Fig. 2. Bloque de construcción para arquitectura ResNet, tomado de [4].

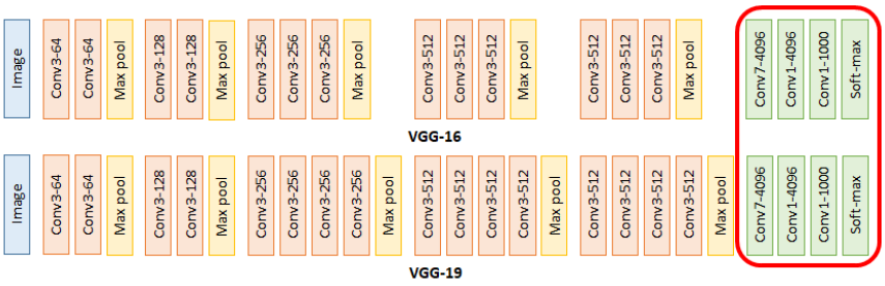


Fig. 3. Arquitectura de una CNN VGG16 y VGG19, tomado de [12].

Entender la causa del sobreajuste es un reto, ya que diseccionar el problema en la práctica requiere una medición precisa de los errores de generalización de las redes neuronales profundas. de las redes neuronales profundas [6].

3. Modelos de redes neuronales convolucionales

Los modelos de redes neuronales convolucionales, CNN por sus siglas en inglés, son arquitecturas muy eficientes, gracias a que cuentan con tres capas principales, las cuales son:

– Capa convolucional.

– Capa de agrupamiento.

– Capa “Fully-Connected”, FC.

La capa convolucional es la primera capa de una red convolucional. Mientras que las capas convolucionales pueden ir seguidas de capas convolucionales adicionales o capas de agrupamiento, la capa totalmente conectada es la última capa. Con cada capa, la CNN aumenta su complejidad, identificando mayores porciones de la imagen. Las primeras capas se centran en características simples, como colores y bordes. A medida que los datos de la imagen avanzan por las capas de la CNN, ésta empieza a reconocer elementos o formas más grandes del objeto hasta que finalmente identifica el objeto deseado, mejorando así su desempeño en la búsqueda y reconocimiento de entradas de audio, diálogo (“speach”), e incluso en imágenes, siendo esta una gran ventaja y área de interés cuando se trata del robustecimiento de algoritmos contra ataques.

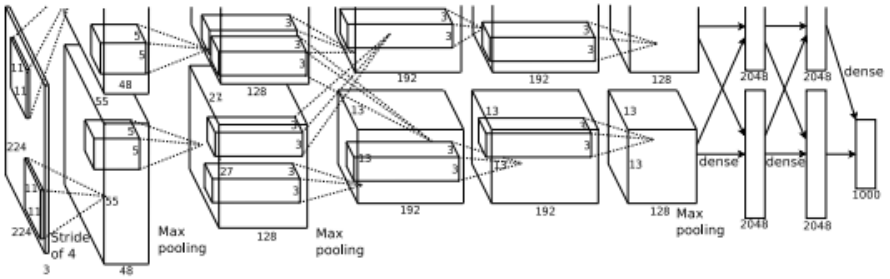


Fig. 4. Ilustración de arquitectura de una CNN, donde muestra explícitamente la delimitación de responsabilidades entre las dos GPU. Una GPU ejecuta las capas de la parte superior de la figura, mientras que la otra ejecuta las capas de la parte inferior. Las GPU sólo se comunican en determinadas capas. tomada de [8].

3.1. ResNet

Residual Network, también conocida como ResNet, es una CNN que acepta una resolución de 224x224, En esta infraestructura en lugar de esperar que cada una de las capas apiladas se ajuste directamente a una cartografía subyacente deseada, se deja explícitamente que estas capas se ajusten a una cartografía residual, Figura 2.

Formalmente, denotando la cartografía subyacente deseada como $\mathcal{H}(x)$, dejando que las capas no lineales apiladas se ajusten a otra cartografía de $\mathcal{F}(x) := \mathcal{H}(x) - x$. El mapeo original se refunde en $\mathcal{F}(x) + x$. Esta infraestructura democratizó el concepto de "residual learning" y "skip connections" mejorando en gran medida la posibilidad de entrenar de manera más profunda los modelos futuros, encontrándose en diferentes presentaciones, siendo de las más conocida:

– ResNet 18.

– ResNet 50.

– ResNet 101.

[NOTA] Estas mismas han sido utilizadas para el análisis de robustez de la arquitectura en este estudio.

3.2. VGG

VGG, Visual Geometry Group, es un arquitectura CNN clásica, que consta de diferentes capas convolucionales y un numero variable, de acuerdo al modelo que mejor ajuste, de capas "fully-connected", siendo lo más común encontrar esta arquitectura con un total de 16 y 19 capas, siendo llamada VGG16 y VGG19 respectivamente. La red VGG se introdujo utilizando capas convolucionales apiladas unas sobre otras a profundidades crecientes. Mediante la agrupación máxima, se reduce el tamaño del volumen. Luego le siguen dos capas totalmente conectadas, cada una con 4.096 nodos y, a continuación, un clasificador softmax.

Tabla 1. Conjuntos de datos de referencia para cambios en la distribución de datos en corrupciones de imágenes en el dataset CIFAR.

Categoría	Dataset	Tipos de variaciones en las imágenes
Corrupción	CIFAR-C	**Ruido** (Gaussiano, Impulso, Disparo, Moteado);
		Borroso (Desenfoque, Cristal, Gaussiano, Movimiento, Zoom);
		Meteorológico (Niebla, Escarcha, Nieve, Salpicaduras);
		Digital (Brillo, Contraste, Elástico, Compresión Jpeg, Pixelado, Saturado)

Tabla 2. Diferencias de precisión obtenidas en modelos Convolucionales a partir de un datatest basado en cifar10-C comparados con un datatest sin corrupciones.

Diferencia de precisión	ResNet18	ResNet50	ResNet101	VGG16	VGG19	AlexNet
Modelo sin preentrenamiento	57 %	58 %	51 %	35 %	37 %	3 %
Modelo con preentrenamiento	85 %	81 %	86 %	81 %	75 %	63 %

VGG en todas sus capas utiliza filtros de convolución muy pequeños (3×3) y el paso convolucional es igual a 1 píxel para reducir el número de parámetros en esta red profunda [12], esta arquitectura se puede apreciar, en el modelo VGG16 y VGG19, en la Figura 3.

3.3. AlexNet

AlexNet es una CNN que contiene ocho capas con pesos; las cinco primeras son convolucionales y las tres restantes están totalmente conectadas. La salida de la última capa totalmente conectada se alimenta a un softmax de 1000 vías que produce una distribución sobre las 1000 etiquetas de clase. La primera capa convolucional filtra la imagen de entrada de $224\times224\times3$ con 96 núcleos de tamaño $11\times11\times3$ con un intervalo de 4 píxeles (ésta es la distancia entre los centros del campo receptivo de las neuronas vecinas en un mapa de núcleos).

La segunda capa convolucional toma como entrada la salida (de respuesta normalizada y agrupada) de la primera capa convolucional y la filtra con 256 núcleos de tamaño $5\times5\times48$. La tercera, cuarta y quinta capas convolucionales están conectadas entre sí sin capas intermedias de agrupación o normalización. La tercera capa convolucional tiene 384 núcleos de tamaño $3\times3\times256$ conectados a las salidas (normalizadas, agrupadas) de la segunda capa convolucional. La cuarta capa convolucional tiene 384 núcleos de tamaño $3\times3\times192$, y la quinta capa convolucional tiene 256 núcleos de tamaño $3\times3\times192$.

Las capas totalmente conectadas tienen 4096 neuronas cada una, Figura 4 [8]: Esta arquitectura permite en gran medida disminuir el “Overfitting”, un comportamiento indeseable del aprendizaje automático que se produce cuando el modelo de aprendizaje automático ofrece predicciones precisas para los datos de entrenamiento, pero no para los datos nuevos.

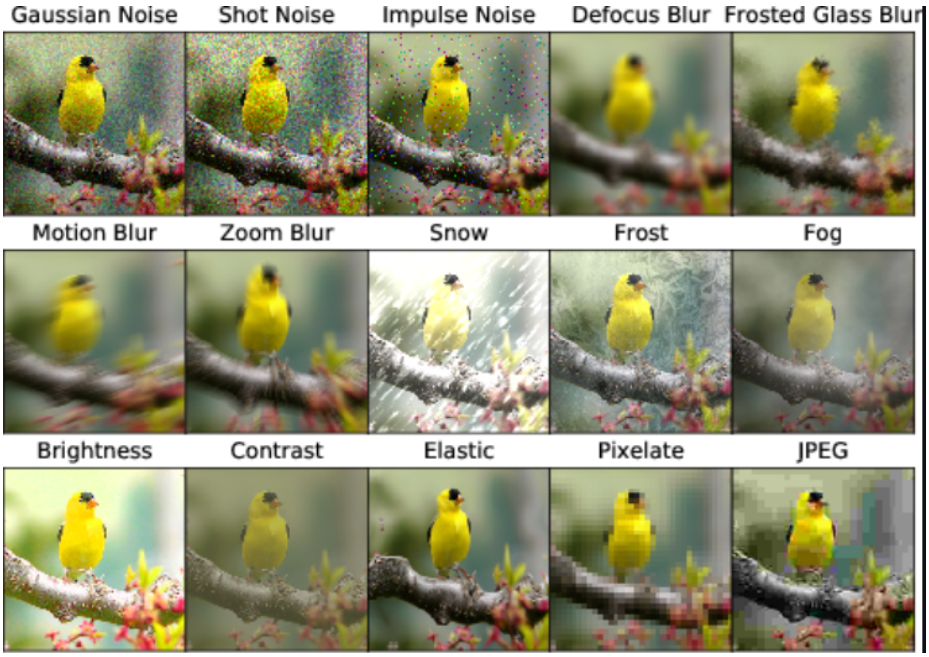


Fig. 5. Ejemplos de corrupciones aplicadas a imágenes, tomado de [5].

4. Conjuntos de datos corruptos

Cifar-C es una variante del conjunto de datos CIFAR, el cual contiene 19 tipos de corrupciones. Sin embargo, este tipo de corrupciones no son suficientes para representar todas las posibles variaciones que pueden ocurrir en el mundo real [15], tabla 1. Sin embargo, estas nos permiten vislumbrar cómo respondería una red neuronal convolucional ante ataques por envenenamiento, también llamados "Poisoning", observando así la precisión, pérdida y robustez de las arquitecturas ante entradas corruptas naturalmente.

5. Comportamiento de arquitecturas CNN ante un dataset corrupto con y sin preentrenamiento

5.1. Configuración de las variables de experimentación

Se siguió la configuración básica para la implementación en Pytorch de un dataset basado en CIFAR-10 dividiendo y descargando la información del datatrain, siguiendo la metodología documentada por [7]; sin embargo para la generación del datatest se prosiguió a partir de las corrupciones documentadas por [5]. Para el punto de comparación se utilizó un el datatest estándar de CIFAR-10.

Tabla 3. Comparación de resultados de pérdida y precisión de los modelos CNN obtenidos a través de fine tunning, con y sin entrenamiento.

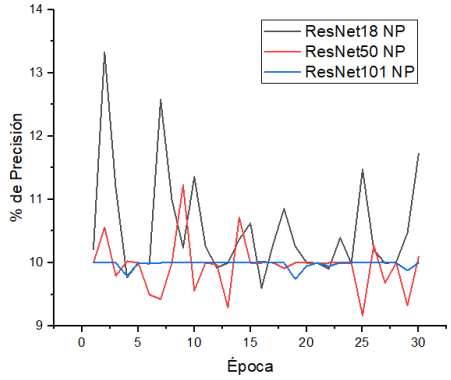
Modelo	Datatest	Pretrain	Loss	Acc
ResNet18	Cifar 10-C	No	4426.91	13.28 %
	Cifar 10-C	Si	780.97	11.24 %
	Cifar 10	No	2.91	31.2 %
	Cifar 10	Si	3.74	75.13 %
ResNet50	Cifar 10-C	No	7065.52	11.22 %
	Cifar 10-C	Si	6799.6	14.57 %
	Cifar 10	No	17	26.87 %
	Cifar 10	Si	3.68	74.8 %
ResNet101	Cifar 10-C	No	3415399	10.01 %
	Cifar 10-C	Si	1017160	10.78 %
	Cifar 10	No	19.22	20.31 %
	Cifar 10	Si	14.33	76.25 %
VGG16	Cifar 10-C	No	434.53	14.44 %
	Cifar 10-C	Si	9047.64	12.91 %
	Cifar 10	No	2.52	22.12 %
	Cifar 10	Si	8.93	68.55 %
VGG19	Cifar 10-C	No	375.85	12.37 %
	Cifar 10-C	Si	843.24	16.85 %
	Cifar 10	No	2.62	19.6 %
	Cifar 10	Si	9.32	68.46 %
ResNet50	Cifar 10-C	No	2.96	13.01 %
	Cifar 10-C	Si	26.98	23.81 %
	Cifar 10	No	2.92	13.37 %
	Cifar 10	Si	11.44	64.98 %

5.2. Modelos de arquitecturas

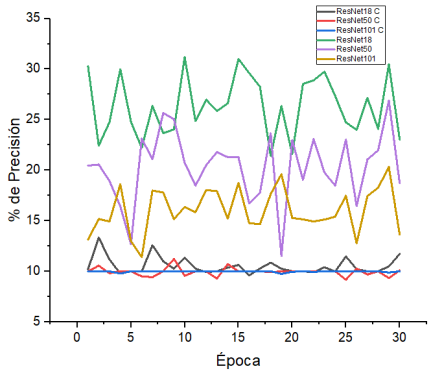
Como base se utilizó el principio de "Fine Tunning" siendo las arquitecturas evaluadas en este experimento: ResNet 18, 50 y 101 [4], así como los modelos VGG16, VGG19 [12] y AlexNet [8].

5.3. Configuraciones del modelo

Se adaptó el modelo de Fine Tunning [1] congelando los parámetros del modelo para que no se actualicen durante el entrenamiento, después se reemplazó la última capa de la red para que tuviera el mismo número de salidas que el número de clases en el conjunto

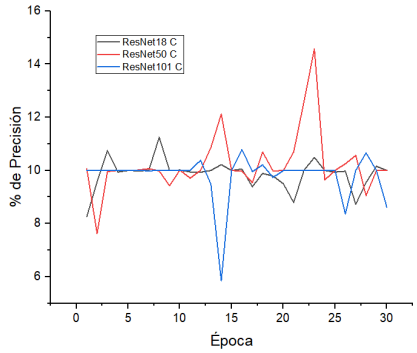


(a) Modelo ResNet18, 50 y 101 sin preentrenamiento (NP).

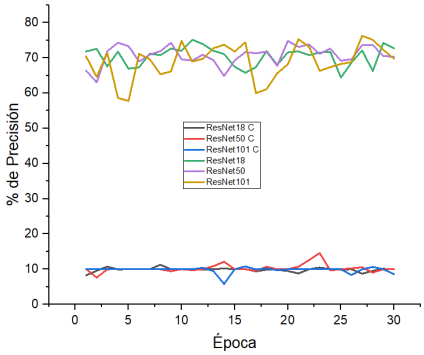


(b) % de precisión del modelo sin pesos de ImageNet del datatest corrupto (ResNetX-C) vs modelo sin pesos del datatest sin corrupciones.

Fig. 6. Porcentaje de precisión de la arquitectura ResNet.



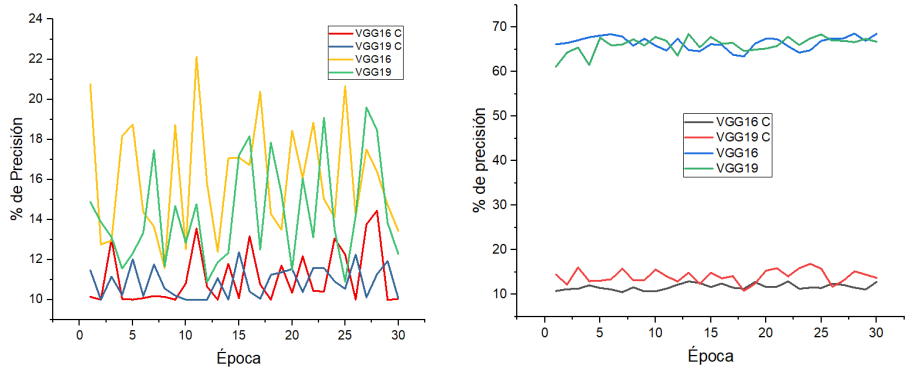
(a) Modelo ResNet18, 50 y 101.



(b) % de precisión del modelo con pesos de ImageNet del datatest corrupto (ResNetX-C) vs modelo del datatest sin corrupciones.

Fig. 7. Porcentaje de precisión de la arquitectura ResNet con preentrenamiento en ImgeNet.

de datos CIFAR10. Cada arquitectura se entrenó en primer término, sin pesos, con un dataset basado en cifar-10, para después evaluar el modelo con el dataset de imágenes corruptas por cifar-10-C, como se puede apreciar en la Fig. 5, evaluando la precisión y pérdida en cada una de las 30 épocas, con Adam como optimizador y Cross Entropy loss como criterion. En segundo término se utilizaron las mismas variables pero con el modelo preentrenado en ImageNet. En tercera instancia se realizaron ambos procesos con un datatest basado en Cifar-10 sin corrupciones, con el objetivo de analizar el gap y comportamiento de los modelos antes mencionados ante entradas con corrupciones naturales y sin ellas.



(a) % de precisión de la CNN VGG16 Y 19 evaluadas sin pesos de ImageNet con Corrupciones (C) y sin ellas.

(b) % de precisión de la CNN VGG16 Y 19 evaluadas cpn preentrenamiento de ImageNet con Corrupciones (C) y sin ellas.

Fig. 8. Porcentaje de precisión de la arquitectura VGG.

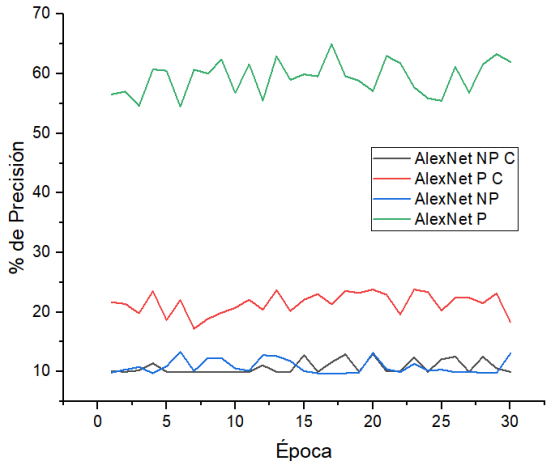


Fig. 9. Precisión obtendia del modelo AlexNet validado sin preeentrenamiento (NP) y con preentrenamiento, en datatest basado en Cifar10 corrupto (c) y sin corrupciones.

6. Resultados

Tras el análisis de las diferentes arquitecturas de redes neuronales, configuradas según lo antes mencionado, en primer lugar para los modelos sin preentrenar, resalta con los mejores resultados el modelo ResNet, en específico la ResNet18, con un porcentaje de precisión del 13.32 %, Figura 7,mientras que la ResNet101 tuvo el peor modelo para este caso, incluso cambiando los parámetros del learning rate, sin embargo, comparados con las validaciones del dataset sin corrupciones se ven claramente opacadas todas las arquitecturas, Fig. 6 inciso b.

Cuando esta arquitectura fue entrenada y evaluada con los pesos de ImageNet, se puede observar una mejoría en la precisión y un descenso en la pérdida, siendo ahora ResNet50 el que mejor responde ante perturbaciones, cuando estos resultados se comparan con sus homólogos sin corrupciones se aprecia que el modelo no tiene problemas, teniendo una precisión promedio del 75.39 %.

Para la Arquitectura VGG, evaluando el modelo sin preentrenamiento se puede observar, Fig.8, que no hay una gran diferencia entre la precisión obtenida con el datatest de CIFAR10-C y su homologo sin corrupciones; En contraste, cuando el modelo se evalúa con los pesos de ImageNet la diferencia entre el modelo con corrupciones es de 68.50 % (promedio) comparado contra el datatest sin ruido.

En la arquitectura AlexNet se puede apreciar, Fig. 9, que el comportamiento de la precisión es muy parecido sin preentrenamiento, tanto con un datatest con corrupciones y sin ellas, además demostró ser la mejor Red neuronal respecto a precisión, llegando a tener 23.81 % exitoso, con unos rangos de perdía muy bajos comparados con las otras arquitecturas.

7. Conclusiones y limitaciones

Con base en los resultados se puede afirmar que conforme a la literatura, los modelos de redes neuronales con menos capas y clases son más robustos antes ataques con corrupciones naturales, ya que al ser más generales los detalles y ruidos tienden a afectar en menor medida la respuesta de la Red Convolucional. Se puede apreciar un patrón de diferencia en las diferentes arquitecturas cuando las entradas tienen corrupciones de tipo natural, por ejemplo, el Modelo ResNet tiene una diferencia promedio del 55 % menor de precisión cuando recibe entradas con ruido natural, en un modelo sin preentrenamiento, y de 86 % menor cuando esta preentrenada comparada con la precisión esperada ante entradas normales, esto sin importar las capas, Tabla 3. Por lo que se podría deducir el impacto en esta métrica conociendo la diferencia de precisión en alguna de sus presentaciones. Cabe aclarar que esta suposición está basada en estas 3 arquitecturas, haría falta corroborar esta hipótesis en CNN más complejas como lo es RaWideResNet-70-16, WideResNet-70-16, entre otras, así como probar diferentes datasets corruptos, en estas mismas, por ejemplo CIFAR100-C, CIFAR10-P, CCC, etc.

Referencias

1. Dhillon, P. S., Foster, D., Ungar, L.: Transfer learning using feature selection (2009) doi: 10.48550/arXiv.0905.4022
2. Drenkow, N., Sani, N., Shpitser, I., Unberath, M.: A systematic review of robustness in deep learning for computer vision: Mind the gap? (2021) doi: 10.48550/arXiv.2112.00639
3. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 1126–1135 (2017)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016) doi: 10.1109/CVPR.2016.90

5. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. In: International Conference on Learning Representations, pp. 1–16 (2019) doi: 10.48550/arXiv.1903.12261
6. Ju, H., Li, D., Zhang, H. R.: Robust fine-tuning of deep neural networks with hessian-based generalization guarantees. In: Proceedings of Machine Learning Research, pp. 10431–10461 (2022)
7. Krizhevsky, A., Hinton, G.: Convolutional deep belief networks on cifar-10 (2010)
8. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, vol. 25, pp. 1–9 (2012)
9. Pan, S. J., Yang, Q.: A survey on transfer learning. IEEE Transactions on knowledge and data engineering, vol. 22, no. 10, pp. 1345–1359 (2009) doi: 10.1109/TKDE.2009
10. Pratt, L., Jennings, B.: A survey of transfer between connectionist networks. Connection Science, vol. 8, no. 2, pp. 163–184 (1996) doi: 10.1080/095400996116866
11. Rice, J. R.: The algorithm selection problem. Advances in computers, vol. 15, pp. 65–118 (1976) doi: 10.1016/S0065-2458(08)60520-3
12. Shadeed, G. A., Tawfeeq, M. A., Mahmoud, S. M.: Automatic medical images segmentation based on deep learning networks. In: IOP Conference Series: Materials Science and Engineering, vol. 870, pp. 012117 (2020) doi: 10.1088/1757-899X/870/1/012117
13. Szegedy, C.: Intriguing properties of neural networks. In: Proceedings of the International Conference on Learning Representations, pp. 1–10 (2013)
14. Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., Schmidt, L.: Measuring robustness to natural distribution shifts in image classification. Advances in Neural Information Processing Systems, vol. 33, pp. 18583–18599 (2020)
15. Wang, S., Veldhuis, R., Strisciuglio, N.: The robustness of computer vision models against common corruptions: A survey (2023) doi: 10.2139/ssrn.4960634
16. Xie, Q., Luong, M. T., Hovy, E., Le, Q. V.: Self-training with noisy student improves imagenet classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10687–10698 (2020) doi: 10.1109/CVPR42600.2020.01070

ISSN 1870-4069

Explorando la robustez adversaria ante el ataque adversario PGD de los modelos AlexNet, VGG y ResNet

María Fernanda Castro-Sandoval[2], Ivan Reyes-Amezcua[1], Andres Mendez-Vazquez[1]

[1] Centro de Investigación y de Estudios Avanzados, Departamento de Computación, Mexico

[2] Universidad de Guadalajara, México

{ivan.reyes, andres.mende}@cinvestav.mx, maria.castro6643@alumnos.udg.mx

Resumen. A pesar de los últimos avances en el campo de Machine Learning y de la aparente precisión y calidad de los modelos actuales de visión por computadora, estos aún preservan una vulnerabilidad, y es ante los ataques adversarios. Estos consisten en ligeras modificaciones (a menudo imperceptibles para el ojo humano) en las imágenes de entrada, capaces de provocar que el modelo haga predicciones incorrectas. En el campo de la robustez adversaria se desarrollan métodos de defensa contra estos ataques, como el entrenamiento adversario y los métodos de generación de ataques como PGD. En este trabajo empleamos estos métodos para poner a prueba la robustez adversaria que pueden alcanzar algunos modelos de referencia como AlexNet, VGG y ResNet sobre dos datasets de referencia como CIFAR-10 y CIFAR-100. Nuestros resultados arrojan que aspectos como el algoritmo de optimización y su tasa de aprendizaje juegan un papel fundamental en el desempeño de los modelos y su robustez adquirida. Además, en nuestros experimentos, el modelo con la menor capacidad fue aquél que logró la mejor precisión tras el entrenamiento adversario, mientras que aquellos con la mayor capacidad presentaron un sobreajuste y porcentajes de precisión más bajos que el modelo de menor capacidad.

Palabras clave: Entrenamiento adversario, ataque adversario, robustez adversaria, PGD, transfer learning finetuning.

Exploring Adversarial Robustness Against PGD Adversarial Attack of AlexNet, VGG, and ResNet Models

Abstract. Despite the recent advances in the field of Machine Learning and the apparent accuracy and quality of current computer vision models, they still preserve a vulnerability, and it is against adversarial attacks. These consist of slight modifications (often imperceptible to the human eye) to input images,

capable of causing the model to make incorrect predictions. In the field of adversarial robustness, defense methods against these attacks are developed, such as adversarial training and attack generation methods like PGD. In this work, we employ these methods to test the adversarial robustness that some reference models like AlexNet, VGG, and ResNet can achieve on two reference datasets like CIFAR-10 and CIFAR-100. Our results show that aspects such as the optimization algorithm and its learning rate play a fundamental role in the performance of the models and their acquired robustness. Additionally, in our experiments, the model with the lowest capacity was the one that achieved the best accuracy after adversarial training, while those with the highest capacity exhibited overfitting and lower accuracy percentages than the model with the lowest capacity.

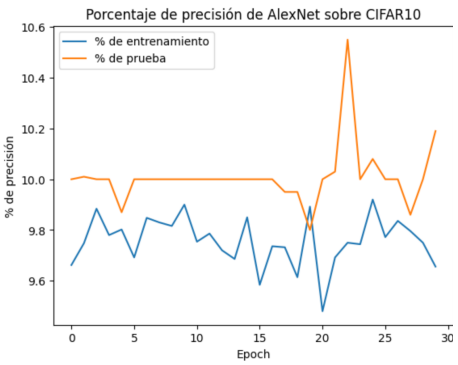
Keywords: Adversarial training, adversarial attack, adversarial robustness, PGD, transfer learning, finetuning.

1. Introducción

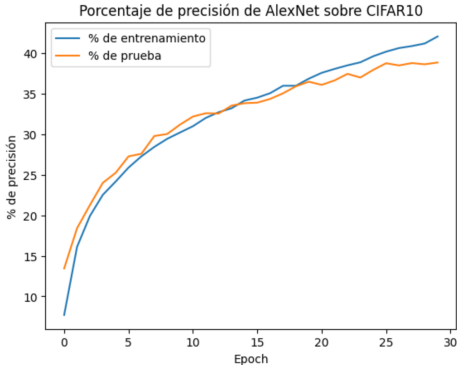
En los últimos años hemos presenciado el auge del campo del Machine Learning, al punto de posicionarse como una tecnología lider. Los modelos de redes neuronales profundas han demostrado ser bastante útiles y eficientes para una gran variedad de aplicaciones. Entre estos, los que resultan más atractivos y apasionantes son aquellos de visión por computadora, que interpretan el mundo visual a través de imágenes y videos para aprender a desempeñar diversas tareas: desde el reconocimiento y generación de imágenes hasta la detección de fraudes, diagnósticos médicos y la conducción de vehículos autónomos. Sin embargo, a pesar de los avances y de la aparente precisión y calidad de los modelos actuales de visión por computadora, existe una problemática importante: estos aún son vulnerables a los ataques adversarios.

Estos ataques consisten en modificaciones mínimas, a menudo imperceptibles para el ojo humano, en las imágenes que conforman el conjunto de datos de entrenamiento del modelo. Aunque pequeñas, estas modificaciones pueden ser capaces de causar que el modelo haga predicciones incorrectas. Dada la creciente aplicabilidad de estos modelos en nuestra vida cotidiana, esta vulnerabilidad no solo cuestiona el nivel de confianza que podemos depositar en estos modelos, sino que también puede dar lugar a problemas más comprometedores en situaciones reales. Por estas razones, el campo de la robustez adversaria, que estudia como mejorar la resistencia en la precisión de los modelos ante los ataques adversarios, se ha convertido en un área de investigación activa.

Entre las estrategias que esta ha desarrollado destaca el entrenamiento adversario, que en pocas palabras consiste en entrenar el modelo incluyendo intencionalmente ejemplos de ataques adversarios en el dataset de entrenamiento. A su vez, se han desarrollado diferentes algoritmos para generar estos ataques intencionados, como Fast Gradient Sign Method (FGSM) ó Basic Iterative Method (BIM). Pero uno de los más importantes y utilizados consiste en la aplicación del método de optimización llamado Projected Gradient Descent(PGD), que se traduce como “Descenso de Gradiente Proyectado”: un algoritmo muy útil para resolver problemas de optimización bajo restricciones.



Optimizador Adam con $\text{lr} = 0.001$:
% de entrenamiento $9.656\,\%$
% de prueba $10.19\,\%$



SGD con $\text{lr} = 0.001$ y $\text{momentum} = 0.9$:
% de entrenamiento $42.06\,\%$
% de prueba $38.85\,\%$

Fig. 1. Comparación en el aprendizaje de AlexNet al cambiar el optimizador Adam por SGD.

Así, las aportaciones de este trabajo consisten en una serie de experimentos de entrenamiento adversario con ataques generados por PGD a diferentes modelos de referencia en el área de visión por computadora: AlexNet, VGG, y ResNet. Asimismo, estos entrenamientos se llevarán a cabo sobre los datasets CIFAR-10 y CIFAR-100, también ampliamente utilizados en este campo.

2. Panorama sobre entrenamiento adversario con PGD

A pesar de que han surgido diferentes algoritmos de primer orden para generar ataques adversarios (i.e. métodos que utilizan información de primer orden sobre el modelo), en [4] se presentó evidencia de que PGD genera ataques "más fuertes" que otros algoritmos de este tipo. Por ejemplo, sus resultados arrojaron que aquellos modelos entrenados con ataques generados por FGSM presentaban un sobreajuste (overfitting) y continuaban siendo vulnerables ante ataques de PGD.

Por el contrario, si se entrena un modelo contra ataques de PGD, este también resistirá ataques generados por muchos otros métodos. Además, estos autores mostraron evidencia empírica sobre la convergencia de PGD hacia el máximo error del modelo para un dato de entrada y la similitud entre los resultados obtenidos al seleccionar distintos puntos iniciales alrededor de este punto de entrada. Finalmente, sus principales experimentos consistieron en el entrenamiento adversario de algunos modelos sobre los datasets MNIST y CIFAR-10.

En particular, sobre CIFAR-10 se entrenó un modelo ResNet y una versión amplificada de este, a la cual se le agregaron capas más amplias por un factor de 10, resultando en un modelo de 5 unidades residuales con (16, 160, 320, 640) filtros cada una. Así, al realizar el entrenamiento adversario con 20 iteraciones de PGD, ℓ_∞ y $\varepsilon = 8$, lograron un porcentaje de precisión de 43.7 % para el modelo original y 45.8 % para su versión amplificada. Una de sus conclusiones consiste precisamente en que la arquitectura de los modelos influye en gran parte sobre la robustez que pueden adquirir,

ya que la frontera de decisión que separa a los datos perturbados puede ser mucho más compleja que aquella que separa a los datos originales; así, los modelos con mayor capacidad (i.e. aquellos que poseen un mayor número de parámetros) presentan un mejor desempeño tras el entrenamiento adversario. En este trabajo vamos a manejar el planteamiento del entrenamiento adversario con PGD que proponen estos autores. Este método de defensa ha sido ampliamente utilizado en la investigación, ya que es de los pocos que entrenan a los modelos para resistir ataques más fuertes. Asimismo, varios trabajos han contribuido a mitigar su mayor desventaja: su alto costo computacional, desarrollando alternativas que logran resultados similares a un costo mucho menor, como Free Adversarial Training (entrenamiento adversario libre) [5], Fast Adversarial Training (entrenamiento adversario rápido) [8], entre otros [7].

3. Metodología y materiales

3.1. Noción matemática de un ataque adversario

Sea $f_\theta : X \to Y$ un modelo de visión por computadora compuesto por redes neuronales profundas con parámetros θ para un conjunto de datos etiquetados $\{X,\ Y\}$ (con $x \in \text{Mat}_{\ell \times w \times 3}(\mathbb{R})$, $\forall x \in X$; $y \in \mathbb{Z}$, $\forall y \in Y$). En este contexto, un ataque adversario para cualquier imagen de entrada $x \in X$ puede expresarse matemáticamente como:

$$x' = x + \delta, \tag{1}$$

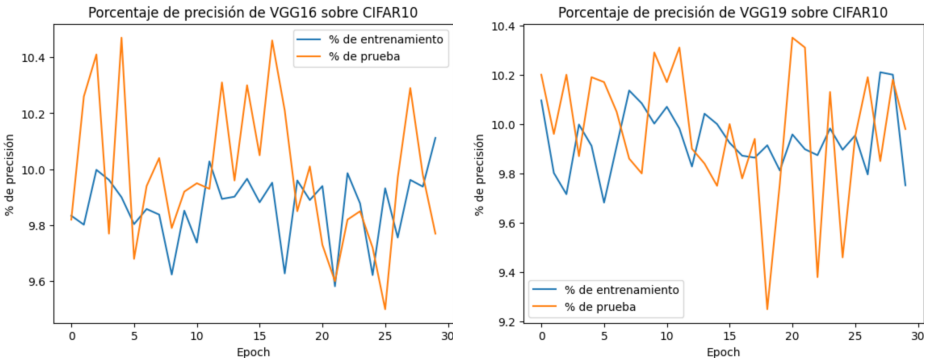
donde δ representa un tensor del mismo tamaño que x (es decir, $\delta \in \text{Mat}_{\ell \times w \times 3}(\mathbb{R})$) cuyos componentes contienen valores numéricos muy pequeños (a menudo denominados como ruido) que al sumarse a los valores de x provocan una ligera perturbación de este, misma que denotamos ahora como x'.

3.2. Entrenamiento adversario

Como se ha mencionado, el campo de investigación de la robustez adversaria estudia y desarrolla estrategias para lograr que los modelos como f_θ mantengan el mayor porcentaje posible de precisión al clasificar este tipo de ejemplos adversarios. Una de las técnicas más utilizadas para este propósito es el Entrenamiento adversario. Este tipo de entrenamiento utiliza Data Augmentation, una técnica que consiste en crear artificialmente datos nuevos a partir de los ya existentes, para aumentar la cantidad de muestras en el conjunto $\{X,\ Y\}$.

En este caso, las nuevas muestras generadas corresponden a un subconjunto de entradas del conjunto original (X) perturbadas mediante ataques adversarios como el de (1). Así, el entrenamiento adversario del modelo f_θ consiste en crear ejemplos adversarios de manera intencionada a partir de los datos de entrada X, para luego agregar estos nuevos ejemplos al mismo conjunto y agregar sus etiquetas (que corresponden a las de los mismos datos sin modificar) al conjunto Y, creando así un nuevo conjunto de datos aumentado al que denotaremos como $\{\bar{X},\ \bar{Y}\}$, sobre el cual finalmente se entrena el modelo.

Optimizador Adam con lr $= 0.001$:



% de entrenamiento 9.9 %
% de prueba 10.47 %

% de entrenamiento 9.958 %
% de prueba 10.35 %

Fig. 2. Resultados de entrenamiento adversario de VGG16 y VGG19 al emplear el optimizador Adam.

Esto se traduce en que, si se tiene la función $L(\theta; x, y)$ que calcula el error del modelo para una sola entrada $x \in X$, resultando en la función de error:

$$J(\theta; X, Y) := \frac{1}{n} \sum_{i=1}^{n} L(f_\theta(x_i), y_i). \tag{2}$$

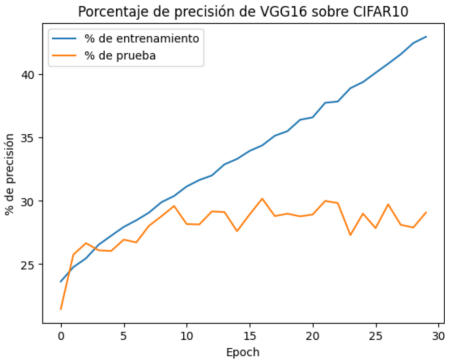
Para entrenar al modelo, entonces durante el entrenamiento adversario surge la nueva función de error:

$$\bar{J}(\theta; \bar{X}, \bar{Y}) := \frac{1}{n+m} \sum_{i=1}^{n} L(f_\theta(x_i), y_i) + \frac{1}{n+m} \sum_{i=1}^{m} L(f_\theta(x_i + \delta_i), y_i), \tag{3}$$

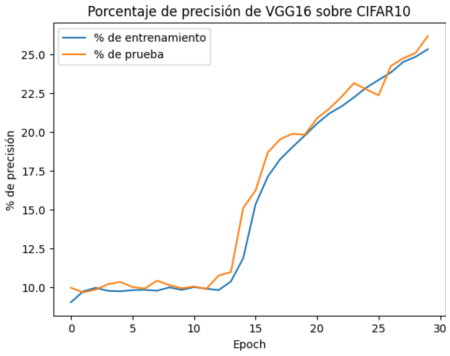
donde n denota el número de entradas en X, m la cantidad de entradas agregadas, $(x_i, y_i) \in X \times Y$ para $i = 1, 2, \ldots, n$, y cada $(x_i + \delta_i, y_i)$ $(i = 1, 2, ..., m)$ corresponde a la entrada (x_i, y_i) modificada.

3.3. Generación de ataques adversarios (PGD)

Existen diferentes técnicas para obtener los ataques adversarios de la forma (1) para aumentar el conjunto de datos $\{X,\ Y\}$. Una de las más importantes es el método PGD. Primero, se mencionó que los valores en el tensor δ son muy pequeños, pero para ser más específicos, se suele elegir a δ de manera que $||\delta||_p \leq \varepsilon$, donde $||\cdot||_p$ (a menudo también denotada por ℓ_p) representa a la norma p con $p \in [1, \infty]$ (así, cuando $p = 2$ corresponde a la norma euclidiana) y ε es un valor positivo suficientemente pequeño en el contexto del problema que se esté resolviendo. De hecho, se puede interpretar a ε como un valor numérico que mide la intensidad de los ataques adversarios, es decir, la diferencia entre una imagen modificada y aquella sin modificar (entre mayor sea ε, mayor será la diferencia).



Optimizador SGD con lr $= 0.1$:
% de entrenamiento 34.38 %
% de prueba 30.17 %



Optimizador SGD con lr $= 0.05$:
% de entrenamiento 25.312 %
% de prueba 26.16 %

Fig. 3. Diferencias en el aprendizaje de VGG16 al cambiar el parámetro lr.

Luego, el valor de ε se mantiene fijo durante todo el entrenamiento adversario, indicando la magnitud de los ataques de los que el modelo está aprendiendo a defenderse. Se pretende que el valor de ε sea lo suficientemente pequeño para que las alteraciones de las imágenes se mantengan imperceptibles para el ojo humano, pero al mismo tiempo logren confundir al modelo. Algunos valores de referencia para ε en la práctica son: $\varepsilon = 0.5$ para ℓ_2, y $\varepsilon = 8/255$, $\varepsilon = 4/255$ para ℓ_∞.

Por otra parte, es razonable pensar que si se lleva a cabo un entrenamiento adversario sobre aquellos ataques que logren la mayor disminución del rendimiento del modelo, la robustez adquirida por este será mayor. Por eso, para un $\varepsilon > 0$ y un par $(x, y) \in X \times Y$ dados, surge la idea de obtener el ataque para x de magnitud menor o igual a ε que provoque el mayor error en la predicción del modelo para esa entrada. Esto se plasma matemáticamente en el problema de optimización:

$$\delta^* = \arg\max_{\delta} L(\theta; f_\theta(x+\delta), y), \quad \text{sujeto a: } ||\delta||_p \leq \varepsilon. \tag{4}$$

Como vemos, este es un problema de optimización con una restricción, de la forma:

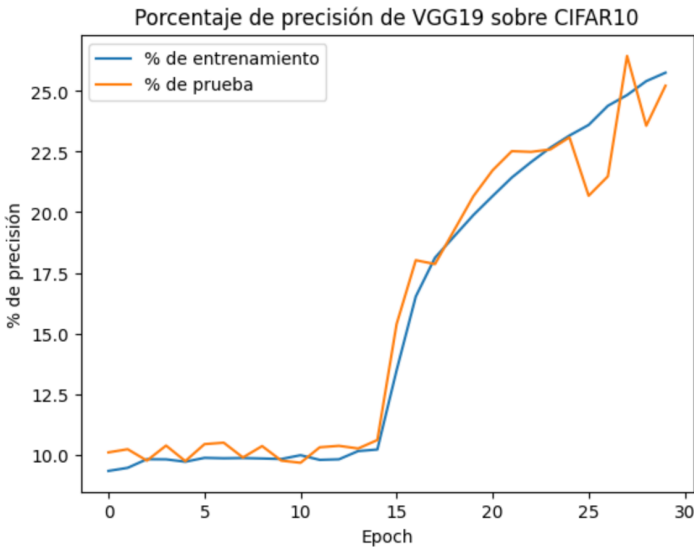
$$\arg\min_{t \in T} g(t), \tag{5}$$

donde $g = -L$, $t = \delta$ y $T = \{\delta : ||\delta||_p \leq \varepsilon\} = B_\varepsilon$, ($B_\varepsilon$ denota a la bola cerrada de radio ε centrada en cero, en $\mathbb{R}^{\ell \times w \times 3}$). Este es el tipo de problemas que resuelve el método del descenso de gradiente proyectado (PGD). Su mecanismo es muy similar al del método del descenso por gradiente, que calcula cada iteración $k+1$ de la forma:

$$t_{k+1} = t_k - \alpha_k \nabla g(t_k). \tag{6}$$

La diferencia consiste en que el método PGD proyecta el resultado de esta iteración hacia el conjunto de la restricción (T), mediante el operador de proyección:

$$\mathcal{P}_T(t_0) := \text{proj}_T(t_0) := \arg\min_{t \in T} ||t - t_0||_p. \tag{7}$$



Optimizador SGD con lr $= 0.05$:
% de entrenamiento 24.834 %
% de prueba 26.45 %

Fig. 4. Resultados del entrenamiento adversario para VGG19.

Además, PGD aplica la función:

$$\begin{aligned} \operatorname{sign}(\boldsymbol{v}) = \operatorname{sign}(v_1, v_2, \ldots, v_n) := (\operatorname{sign}(v_1), \operatorname{sign}(v_2), \ldots, \operatorname{sign}(v_n)), \\ \boldsymbol{v} \in \mathbb{R}^n,\ n \in \mathbb{N}. \end{aligned} \tag{8}$$

Al gradiente $\nabla g(t_k)$. De este modo, cada iteración $k+1$ del método del descenso de gradiente proyectado tiene la forma:

$$t_{k+1} = \mathcal{P}_T[t_k - \alpha_k \operatorname{sign}(\nabla g(t_k))]. \tag{9}$$

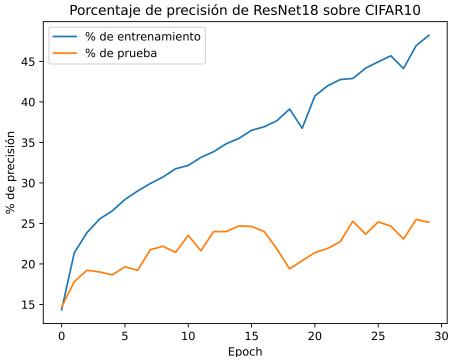
Así, para el caso que nos interesa (4), esta tiene la forma:

$$\delta_{k+1} = \mathcal{P}_{B_\varepsilon}[\delta_k + \alpha_k \operatorname{sign}(\nabla_\delta L(\theta; f_\theta(x+\delta), y))]. \tag{10}$$

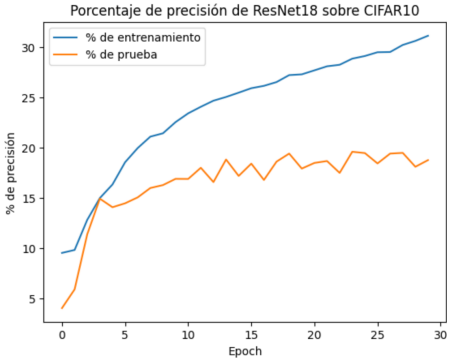
De esta manera se calcula cada iteración hasta que se cumpla algún criterio de paro, resultando en la δ^* que se buscaba en (4). En la práctica se suele elegir una cantidad fija de iteraciones, ya que de estas también puede depender la magnitud del ataque (entre más iteraciones, más cerca se puede llegar a aquel δ^* que maximiza el error del modelo). La elección del valor inicial δ_0 y cada valor α_k también varía según el problema. Una vez creados los ataques adversarios con este método, y agregados al dataset $\{X,\ Y\}$ formando uno nuevo: $\{\bar{X},\ \bar{Y}\}$, lo siguiente es entrenar al modelo f_θ sobre $\{\bar{X},\ \bar{Y}\}$, es decir, resolver el problema:

$$\theta^* := \arg\min_\theta \left(\frac{1}{n+m} \sum_{i=1}^{n} L(f_\theta(x_i), y_i) + \frac{1}{n+m} \sum_{i=1}^{m} L(f_\theta(x_i + \delta_i^*), y_i) \right). \tag{11}$$

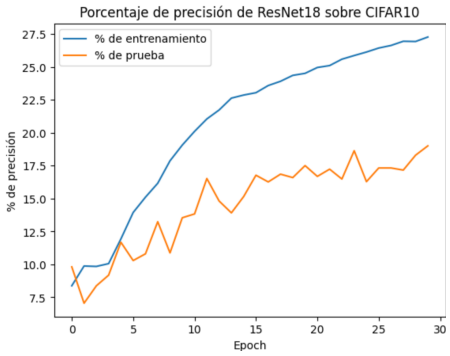
De este modo se lleva a cabo un entrenamiento adversario aplicando ataques de PGD.



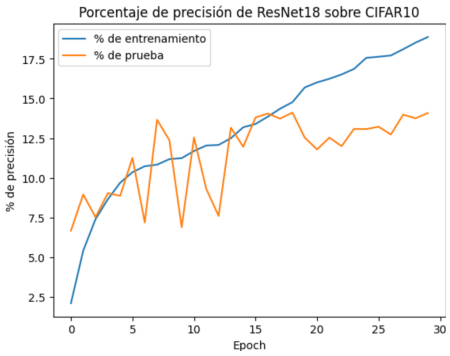
Optimizador Adam con lr $= 0.001$:
% de entrenamiento 46.956 %
% de prueba 25.49 %



Optimizador SGD con lr $= 0.1$:
% de entrenamiento 28.84 %
% de prueba 19.59 %



Optimizador SGD con lr $= 0.05$:
% de entrenamiento 27.264 %
% de prueba 19.0 %



Optimizador SGD con lr $= 0.001$:
% de entrenamiento 14.766 %
% de prueba 14.766 %

Fig. 5. Experimentos realizados para el modelo ResNet18.

3.4. Transfer learning

La cantidad de ejemplos incluidos en un dataset, así como su estructura y distribución, son aspectos fundamentales para una mayor calidad en el aprendizaje de un modelo. Sin embargo, en la práctica no siempre es sencillo reunir la suficiente cantidad de datos, ya que tanto su recolección como su almacenamiento pueden llegar a ser tareas muy costosas; sobretodo si los datos consisten en imágenes.

En el campo de la visión por computadora se han desarrollado varias bases de datos de código abierto suficientemente extensas y completas. Una de las más grandes y ampliamente utilizadas es conocida como ImageNet, aunque en realidad, la mayoría de las veces se refiere con este nombre a un "pequeño" subconjunto de ella (que aún es muy extenso) con más de 1.2 millones de imágenes clasificadas en 1,000 categorías correspondientes a distintos animales y objetos. Este subconjunto recibe el nombre de ImageNet Large Scale Visual Recognition Challenge (ILSVRC), que se traduce como "Desafío de reconocimiento visual a gran escala de ImageNet".

Su nombre se debe a que, entre 2010 y 2017, el proyecto ImageNet organizaba un concurso anual en el que se entrenaban modelos de clasificación de imágenes sobre este. Varios de los modelos ganadores de esta competencia —que entrenaremos en este trabajo– marcaron grandes pasos en la historia de este campo y se convirtieron en modelos de referencia para redes neuronales convolucionales. En la siguiente subsección se describirán más a detalle.

Además de contribuir a una buena precisión en los modelos entrenados, se sabe que el tamaño y contenido de ILSVRC es ideal para que estos mismos modelos muestren un buen desempeño sobre otros datasets con categorías similares. Sin embargo, tal cantidad de contenido también provoca que el entrenamiento sobre este dataset resulte cuando menos desafiante, además de demandar un equipo de hardware especializado.

No obstante, hoy en día se encuentran disponibles en diferentes bibliotecas de software (como Pytorch) o distintos foros y repositorios en línea los pesos que se han obtenido después de entrenar a estos modelos sobre ILSVRC. Así, es posible obtenerlos y emplearlos como pesos iniciales (en lugar de generar a estos últimos aleatoriamente) para entrenar los mismos modelos sobre nuestro dataset más pequeño pero de naturaleza similar. A esta práctica se le conoce como transfer learning (aprendizaje por transferencia), y más específicamente, corresponde al método de Finetuning. En este trabajo se aplicará esta estrategia para realizar un entrenamiento adversario de los modelos que se mencionan a continuación sobre datasets más pequeños, que a su vez se describen en la sección 3.6.

3.5. Modelos

En este caso se optó por experimentar con 3 tipos diferentes de modelos, todos ganadores del ILSVRC: en un principio AlexNet, luego los modelos VGG, y finalmente los modelos de redes neuronales residuales o ResNet. En seguida se profundiza en cada una de sus arquitecturas.

AlexNet. En 2012 se popularizaron las redes neuronales convolucionales gracias al surgimiento de AlexNet: una red de este tipo compuesta por cinco capas convolucionales y tres capas totalmente conectadas (fully connected) que ganó la competencia de ILSVRC en ese mismo año. Recibe su nombre gracias a su creador, Alex Krizhevsky [3].

VGG. Su nombre proviene de las siglas para Visual Geometry Group, que se traduce como "Grupo de Geometría Visual". Este modelo propuesto por K. Simonyan y A. Zisserman de la Universidad de Oxford ganó el segundo lugar en la competencia de ILSVRC en el año 2014 [6]. Existen dos versiones de este, conocidas como VGG16 Y VGG19. Mediante su creación se experimentó en el aspecto de la profundidad en modelos de redes neuronales convolucionales, ya que ambas versiones poseen arquitecturas más profundas que sus predecesoras, con 16 y 19 capas en total respectivamente.

ResNet. Es común que en la práctica se espere que aquellos modelos con mayor profundidad sean los que reflejen el mayor desempeño. Sin embargo, la práctica de apilar una gran cantidad de capas de manera "ingenua" en un modelo puede dar lugar a algunos inconvenientes, como el desvanecimiento del gradiente y el sobreajuste.

Tabla 1. Resultados del entrenamiento adversario de los 3 modelos ResNet sobre CIFAR-10, con el optimizador SGD y lr $= 0.05$.

	ResNet18	**ResNet50**	**ResNet101**
% de entrenamiento	27.264 %	36.658 %	35.862 %
% de prueba	19.0 %	16.16 %	25.67 %

Por esta razón, los creadores de ResNet (el equipo de Microsoft Research liderado por Kaiming He) desarrollaron una manera ingeniosa de conectar las capas de una red profunda evitando estos inconvenientes: mediante los llamados "bloques residuales". Un bloque residual consiste en un par de capas de pesos sinápticos con una función de activación ReLU, donde, además de las conexiones directas, existe una conexión atajo que traslada el valor de entrada x y lo suma al valor resultante de aplicar las capas anteriores ($\mathcal{F}(x)$). Esto favorece sobretodo a la propagación del gradiente. Hasta ahora se han desarrollado varias arquitecturas diferentes de este modelo. El ganador de la competencia ILSVRC en 2015 contiene en total 152 capas [1]. En este trabajo nos enfocaremos en las versiones: ResNet18, ResNet50 y ResNet101, que contienen 18, 50 y 101 capas de profundidad respectivamente.

3.6. Datasets

Los datasets sobre los que vamos a trabajar también son muy utilizados en el área de visión por computadora, aunque son mucho más pequeños que ImageNet. Se describen a continuación.

CIFAR-10: Su nombre corresponde a las siglas para Canadian Institute for Advanced Research, y el número 10 se debe a que posee 10 clases. En realidad es un subconjunto de otro dataset mucho más extenso, llamado 80 million tiny images (80 millones de imágenes diminutas) recolectado por Alex Krizhevsky junto con Vinod Nair y Geoffrey Hinton [2]. CIFAR-10 contiene 6,000 imágenes por cada clase, de las cuales 5,000 pertenecen al conjunto de entrenamiento y 1,000 al conjunto de prueba.

Así, en total posee 60,000 imágenes de 32×32 pixeles cada una, donde los conjuntos de entrenamiento y de prueba están formados por 50,000 y 10,000 imágenes respectivamente.

CIFAR-100: Este dataset proviene de la misma fuente que el anterior y posee el mismo tamaño; la diferencia consiste en que este contiene 100 categorías de clasificación. Así, sus 60,000 imágenes se reparten en 600 para cada clase, con 500 en el conjunto de entrenamiento y 100 en el conjunto de prueba. Sus 100 categorías se dividen en 20 grupos llamados "superclases", donde cada superclase contiene 5 clases. De este modo, a cada imagen del dataset se le asignan 2 etiquetas: una etiqueta "fina" que indica la clase a la que pertenece, y una etiqueta "gruesa" que corresponde a su superclase.

4. Planteamiento de los experimentos y resultados obtenidos

Todos los experimentos incluidos en este trabajo se han realizado con la biblioteca Pytorch y consisten en el entrenamiento adversario de versiones preentrenadas de los modelos (es decir, empleando el método de finetuning, al tomar como pesos iniciales

Tabla 2. Resultados del entrenamiento adversario de todos los modelos sobre ambos datasets.

	CIFAR-10		**CIFAR-100**	
	% de entrenamiento	% de prueba	% de entrenamiento	% de prueba
AlexNet	42.06 %	38.85 %	17.022 %	15.37 %
VGG16	25.312 %	26.16 %	14.224 %	12.23 %
VGG19	24.834 %	26.45 %	9.862 %	9.19 %
ResNet18	27.264 %	19.0 %	11.046 %	4.61 %
ResNet50	36.658 %	16.16 %	19.43 %	3.41 %
ResNet101	35.862 %	25.67 %	33.56 %	6.37 %

a aquellos obtenidos en el entrenamiento sobre ImageNet). Todos los entrenamientos constan de 30 épocas. En cada una de ellas, las imágenes del dataset se agrupan en mini-batches (en pequeños lotes) y cada uno de ellos recibe un ataque de PGD. Estos ataques se generaron mediante 10 iteraciones, utilizando un valor constante $\alpha_k = 0.01$ y partiendo de $\delta_0 = \vec{0}$. En cuanto a la magnitud del ataque, para todos los experimentos se consideró $\boldsymbol{\varepsilon = 0.5}$. Finalmente, en nuestros experimentos, los ataques adversarios tomaron el lugar de cada imagen en cada mini-batch, formando así un dataset constituido puramente por imágenes perturbadas, como se sugiere en [4].

En este nuevo dataset es donde calculamos la función de error y los pasos de optimización. Para entrenar el modelo AlexNet se tuvo que cambiar la dimensión de las imágenes, así como aplicarles cortes centrales. Además, se normalizaron los datos utilizando los valores [0.485, 0.456, 0.406] para la media y [0.229, 0.224, 0.225] para la desviación estándar. Mientras tanto, para el resto de los modelos únicamente se normalizaron los datos con valores de 0.5 para la media y la desviación estándar. En seguida se presentan varios resultados obtenidos en estos experimentos para cada dataset.

4.1. Resultados para CIFAR-10

Uno de los detalles más importantes que se notó en los experimentos fue que el aprendizaje de los modelos era muy susceptible a la arquitectura elegida para el entrenamiento. En la figura 1 se ilustra esta situación para el modelo Alexnet: en un inicio se llevó a cabo su entrenamiento adversario con el optimizador Adam, pero su aprendizaje se estancaba y sus porcentajes de precisión se mantenían al rededor del 10 %. Sin embargo, al cambiar el optimizador por Stochastic Gradient Descent (SGD) y agregar el parámetro momentum $= 0.9$, su aprendizaje se mantuvo mayormente ascendente para ambos conjuntos de datos, llegando a precisiones de alrededor del 40 %.

Lo mismo ocurrió al entrenar los modelos VGG16 y VGG19 con el optimizador Adam y el mismo valor para la tasa de aprendizaje o learning rate (lr), como se muestra en la Figura 2. Para VGG16, al igual que en el caso anterior, se optó por cambiar el optimizador por SGD. Sin embargo, aunque la curva de aprendizaje lucía mayormente creciente, los porcentajes de precisión solo llegaron a 9.724 % para el conjunto de entrenamiento y 10.11 % para el conjunto de prueba.

Por esta razón, se optó por elegir una tasa de aprendizaje de mayor magnitud, en este caso 0.1. Aunque los resultados mejoraron, ocurrió un sobreajuste del modelo. Así, en seguida se probó con una tasa de aprendizaje ligeramente más pequeña: 0.05. Sorprendentemente, este cambio eliminó el sobreajuste y mantuvo el aprendizaje mayormente ascendente, aunque bajando un poco los porcentajes de precisión, como se muestra en la Figura 3.

No obstante, consideramos satisfactorios a estos últimos resultados obtenidos para VGG16, por lo que se decidió emplear la misma arquitectura de entrenamiento para VGG19 (optimizador SGD con lr $= 0.05$). Como vemos en la Figura 4, para este último modelo se obtuvieron resultados bastante similares a los de VGG16. Luego, para el caso de ResNet18, se realizaron experimentos con las 4 arquitecturas que arrojaron resultados satisfactorios en los modelos anteriores (Figura 5). Sin embargo, como podemos observar, en cada uno de ellos ocurrió un sobreajuste del modelo.

El experimento donde el sobreajuste fue menor es aquél donde se empleó el optimizador SGD con lr $= 0.05$ (el experimento con lr $= 0.001$ fue descartado debido a que refleja un aprendizaje lento, en el cual, quizá con un mayor número de épocas, se llegue a un sobreajuste similar al del resto de los casos). Por esta razón, para los modelos ResNet50 y ResNet101 se llevó a cabo un solo experimento con SGD y lr $= 0.05$. En la tabla 1 se presenta una comparación entre los porcentajes obtenidos por los 3 modelos ResNet con esta arquitectura de entrenamiento. En ella podemos notar que el sobreajuste se presenta en los 3 modelos, siendo ResNet50 el más afectado por este fenómeno.

4.2. Resultados para CIFAR-100

Dada la similitud entre ambos datasets, para realizar los entrenamientos de los modelos sobre CIFAR-100 se optó por emplear las mismas arquitecturas de entrenamiento que resultaron adecuadas para el dataset anterior. Así, todos los modelos fueron optimizados por SGD, y los valores de lr se mantuvieron para cada modelo: AlexNet con 0.001 y momentum $= 0.9$; y el resto de los modelos con 0.05. En la tabla 2 se presenta en resumen todos los porcentajes de precisión obtenidos en los experimentos anterior mencionados sobre CIFAR-100, junto con todos aquellos obtenidos sobre CIFAR-10 a modo de comparación. Nótese que los modelos ResNet aún conservan el sobreajuste sobre CIFAR-100, además de que, en general, las precisiones de todos los modelos son más bajas para este último dataset.

5. Discusión

De todos nuestros resultados podemos concluir que el algoritmo de optimización empleado durante el entrenamiento adversario juega un papel fundamental en el desempeño de cada modelo. En particular, el algoritmo Adam resultó inadecuado para todos los casos, mientras que PGD demostró levantar las curvas de aprendizaje y una posible convergencia (coincidiendo con el método propuesto en [4]). Asimismo, el valor de la tasa de aprendizaje también resultó ser muy relevante para los resultados, siendo 0.05 el más adecuado para la gran mayoría de los experimentos.

El sobreajuste de los modelos ResNet puede deberse a la profundidad y complejidad de sus estructuras, en comparación con AlexNet: el modelo con menor cantidad de capas que los demás y al mismo tiempo aquél que arrojó los mejores resultados. Asimismo, resulta razonable que las precisiones obtenidas sobre el dataset CIFAR-100 sean significativamente menores, además de haber obtenido un sobreajuste mucho más pronunciado en los modelos ResNet. Esto se debe a que, al tener un dataset con la misma cantidad de imágenes, pero un número de clases 10 veces más grande, la tarea de clasificación se torna mucho más complicada de llevar a cabo. Una complementación a este trabajo podría consistir en la realización de los mismos experimentos pero con una mayor cantidad de épocas, con el objetivo de observar la convergencia del aprendizaje. Asimismo, surge el interés de estudiar la relación entre la capacidad de los modelos y su desempeño tras el entrenamiento adversario, dado que en este caso una red AlexNet logró obtener un mejor aprendizaje que un modelo ResNet.

Referencias

1. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)
2. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images (2009)
3. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM, vol. 60, no. 6, pp. 84–90 (2017) doi: 10.1145/3065386
4. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv, vol. 1050, no. 9 (2017)
5. Shafahi, A., Najibi, M., Ghiasi, M. A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., Goldstein, T.: Adversarial training for free! Advances in neural information processing systems, vol. 32 (2019)
6. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv, (2014)
7. Sriramanan, G., Addepalli, S., Baburaj, A.: Towards efficient and effective adversarial training. Advances in Neural Information Processing Systems, vol. 34, pp. 11821–11833 (2021)
8. Wong, E., Rice, L., Kolter, J. Z.: Fast is better than free: Revisiting adversarial training. arXiv, (2020) doi: 10.48550/arXiv.2001.03994

ISSN 1870-4069

Clasificador de bosquejos utilizando memoria asociativa entrópica pesada

Julian Rodrigo González-Hernández[1], Karina M. Figueroa-Mora[1], Luis A. Pineda[2], Rafael Morales-Gamboa[3]

[1] Universidad Michoacana de San Nicolás de Hidalgo, Mexico

[2] Instituto de Investigación en Matemáticas, Universidad Nacional de México, Mexico

[3] Universidad de Guadalajara, Mexico

karina.figueroa@umich.mx, lpineda@iimas.unam.mx, rmorales@suv.udg.mx

Resumen. La memoria asociativa entrópica es un concepto relacionado con la capacidad de un sistema para almacenar y recuperar información de manera eficiente y confiable, incluso cuando las entradas son parciales o ruidosas. Utilizar este tipo de memoria ayuda a aprovechar eficientemente la memoria de un sistema, pues almacena los datos de manera compacta y los asocia inteligentemente, lo cual es muy útil en sistemas con recursos limitados. En este artículo se presenta un clasificador de bosquejos hechos a mano (fuente Quick, Draw! de Google Creative Lab) usando redes neuronales convolucionales como módulo de entrada y salida de información, implementando además un sistema de memoria asociativa entrópica pesada con el objetivo de almacenar y recuperar los elementos de la base de datos (trazos). En este artículo se muestran experimentos que exploran las configuraciones de memoria propuestas. Finalmente, se presenta la configuración óptima de la memoria para esta base de datos con respecto a las clases seleccionadas.

Palabras clave: Memoria asociativa entrópica, clasificador de imágenes, redes neuronales convolucionales.

Sketch Classifier using Weighted Entropic Associative Memory

Abstract. Entropic associative memory is a concept related to the ability of a system to store and retrieve information efficiently and reliably, even when the inputs are partial or noisy. Using this type of memory helps to efficiently take advantage of a system's memory, as it stores data compactly and associates it intelligently, which is very useful in systems with limited resources.

This article presents a hand-drawn sketch classifier (source Quick, Draw! from Google Creative Lab) using convolutional neural networks as an information input and output module. It also implements a Weighted entropy associative memory system to store and retrieve database elements (sketch). This article shows experiments exploring the proposed memory configurations. Finally, the optimal memory configuration for this database concerning the selected classes is presented.

Keywords: Entropic associative memory, images classifier, convolutional neural networks.

1. Introducción

Las memorias naturales de los seres humanos y otros animales con un sistema neuronal suficientemente desarrollado son asociativas [6]. Así, por ejemplo una imagen, un sabor o un olor pueden iniciar una cadena de recuerdos a partir de sus significados o contenidos. En otras palabras, estos estímulos se encuentran asociados con recuerdos, los cuales podrían representan información guardada en la memoria. El acceso a esta información es a través de señales, claves o descripciones, además la recuperación de la memoria es una operación constructiva, esto es un recuerdo puede servir de pista para recordar mas elementos adicionales por esto es importante su estudio.

En [6] los autores reportan que ha sido extremadamente difícil crear modelos computacionales de memorias asociativas dentro del paradigma simbólico, y aunque se han realizado importantes intentos de utilizar redes semánticas [9] desde el inicio y sistemas de producción más recientemente [1], aún hacen falta memorias asociativas simbólicas prácticas. Algunos trabajos han estudiado este tipo de memoria dentro del paradigma de las redes neuronales [10]. Esto es, almacenan una gran cantidad de patrones que pueden seleccionarse con señales completas o parciales, así como recuperar el patrón que está más cerca de la pista según una función abstracta.

Las posibles heurísticas usadas para representar la memoria reflejan estrategias análogas utilizadas en representaciones simbólicas donde la negación se equipara con la falta de prueba. Sin embargo, en este artículo se utiliza una tabla para representar la memoria asociativa como se describe en [5]. De manera general, el aporte de este artículo consiste en mostrar la configuración óptima de los parámetros para la implementación de una memoria asociativa entrópica pesada para la clasificación de bosquejos hechos a mano usando redes neuronales convolucionales como mecanismo de entrada/salida de información. Es importante resaltar que la red neuronal también se utilizará para comprobar el resultado de la memoria entrópica.

2. Estado del arte

El primer trabajo publicado sobre memorias asociativas entrópicas se puede consultar en [6]. En este los autores presentan la idea y cómo es posible usar este tipo de memoria para la recuperación de información.

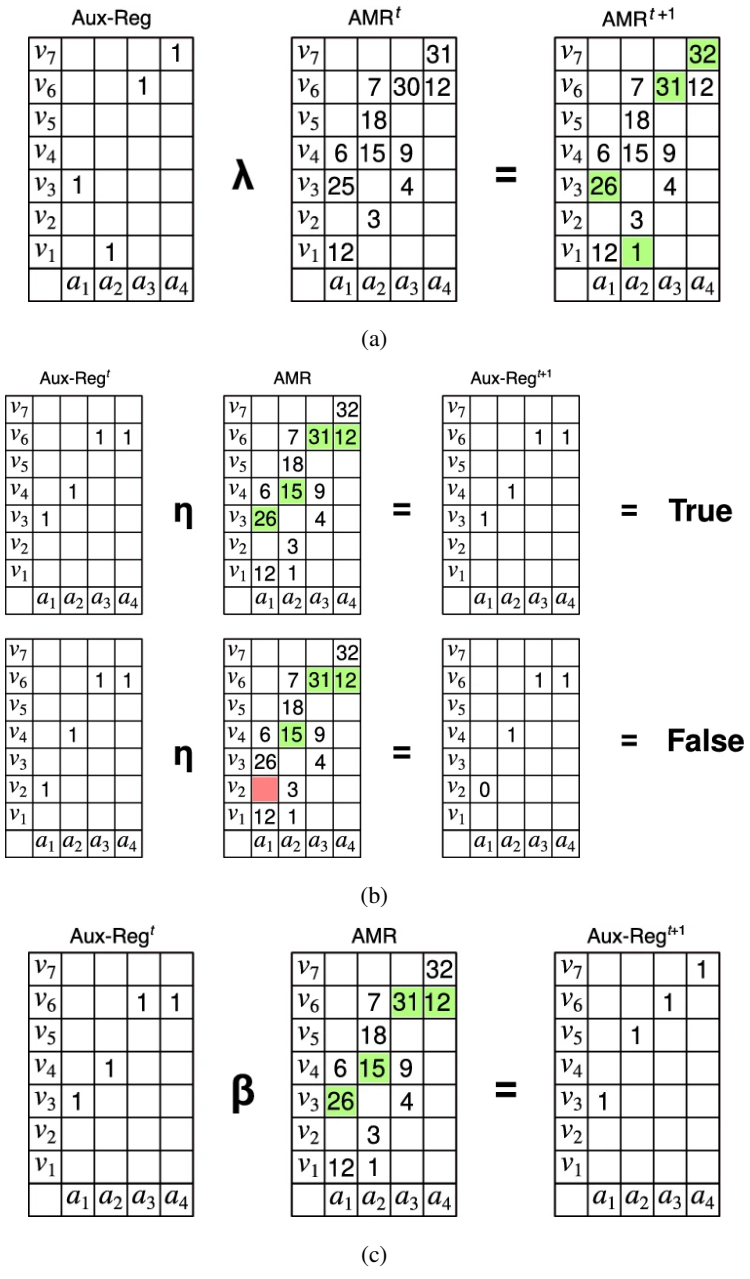


Fig. 1. Operaciones de la memoria asociativa entrópica pesada [8].

En este primer artículo los autores presentan experimentos relacionados con el reconocimiento de dígitos escritos a mano utilizando el corpus MNIST, en el cual se obtuvieron resultados muy favorables con una memoria de tamaño 64×64 presentando 100 % de precisión, 95 % de recuperación y con una entropía promedio de 4.2.

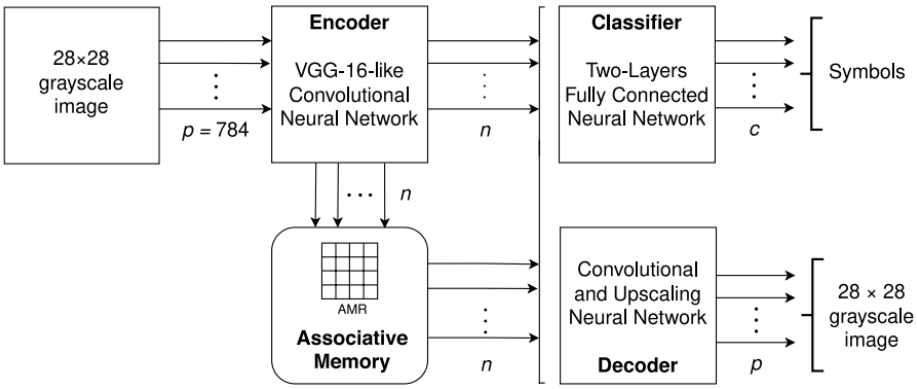


Fig. 2. Arquitectura de la memoria asociativa pesada [8].

Posteriormente, en [4] se presenta el uso de la memoria asociativa entrópica para la detección de símbolos manuscritos utilizando en ésta ocasión el corpus EMNIST donde los autores determinaron las características eficientes de la memoria para ese dominio; en éste caso la memoria de tamaño 64×128 con 87 % de precisión, 77 % de recuperación y una entropía promedio de 5.4. En [7] los autores introducen la idea de memoria asociativa entrópica pesada y muestran un caso de estudio de la representación y el aprendizaje fonético. En este artículo se muestran los experimentos para concluir los parámetros eficientes en el uso de este tipo de memoria específicamente en ese dominio utilizando el corpus de DIME×100; se muestra que el mejor tamaño de los registros de memoria es 32×32 en este caso.

3. Marco teórico

La memoria asociativa entrópica (EAM, por sus siglas en inglés) es un concepto que busca unir los procesos de memoria natural con los modelos computacionales. Los sistemas de EAM están definidos mediante el modo de computación relacional indeterminada (RIC, por sus siglas en inglés).

3.1. Computación relacional indeterminada

La RIC es un enfoque que combina la estructura relacional con cierta flexibilidad o ambigüedad en la interpretación de los datos. El objeto de computación básico de este modo es la relación matemática, entre un objeto en el dominio con uno ó varios objetos del co-dominio [6]. Sean los conjuntos $A = \{a_1, ..., a_n\}$ y $V = \{v_1, ..., v_m\}$, con cardinalidades n y m, el dominio y co-dominio de una relación finita $r : A \to V$, donde los objetos del dominio son llamados argumentos y los del co-dominio son llamados valores. Por cuestiones de notación para cualquier relación *r* se define una función $R : A \times V \to \{0, 1\}$ tal que $R(a_i, v_j) = 1$ si el argumento a_i se relaciona con el valor v_j y $R(a_i, v_j) = 0$ en caso contrario.

Fig. 3. Ejemplo los resultados al concluir el juego de Quick Draw! [3].

La RIC tiene una representación extensional, es decir, se centra en enumerar ejemplos concretos y propiedades generales de los conceptos, además sus algoritmos son indeterminados y entrópicos con una arquitectura distribuida y paralela.

Entropía Computacional La entropía computacional ó entropía de la información, introducido por Claude Shannon en 1948 [11], cuantifica la incertidumbre de variables aleatorias y puede ser utilizada para medir la indeterminación en una relación matemática. Por ejemplo, si la entropía de una función es cero, es porque está completamente determinada, y sería máxima cuando todos los elementos del dominio están relacionados con todos los del co-dominio.

Operaciones Relacionales En términos generales, el RIC se caracteriza por tres operaciones fundamentales: abstracción, inclusión y reducción. Consideremos dos relaciones arbitrarias, r_f y r_a, que mapean elementos desde un conjunto A a un conjunto V con una función f_a sobreyectiva. Es importante destacar que una función también es una relación, por lo que en las siguientes definiciones, las relaciones pueden ser funciones. Estas operaciones, de acuerdo a [5], se definen como sigue:

– **Abstracción:** $\lambda(r_f, r_a) = q$, tal que $Q(a_i, v_j) = R_f(a_i, v_j) \vee R_a(a_i, v_j)$ para todo $a_i \in A$ y $v_j \in V$, es decir, $\lambda(r_f, r_a) = r_f \cup r_a$. Es decir, construye una relación a partir de dos relaciones.

Tabla 1. Categorías utilizadas en el segundo conjunto de experimentos.

Categoría	Etiqueta	Categoría	Etiqueta	Categoría	Etiqueta	Categoría	Etiqueta
Avión	0	Manzana	5	Ave	10	Rana	15
Reloj despertador	1	Espárrago	6	Automóvil	11	Caballo	16
Ambulancia	2	Hacha	7	Gato	12	Calavera	17
Hormiga	3	Mochila	8	Perro	13	Camión	18
Yunque	4	Banana	9	Pato	14	Brazo	19

Fig. 4. Ejemplos de las imágenes obtenidas de Quick, Draw!

– **Inclusión:** $\eta(r_a, r_f)$ es verdadero si $R_a(a_i, v_j) \rightarrow R_f(a_i, v_j)$ para todo $a_i \in A$ y $v_j \in V$ (es decir implicación material), y falso en caso contrario.

– **Reducción:** $\beta(f_a, r_f) = f_v$ tal que si $\eta(f_a, r_f)$ es verdad $f_v(a_i) = r_f(a_i)$ para todo a_i, donde la distribución aleatoria se centra en f_a con varianza σ. Si $\eta(f_a, r_f)$ no se satisface $\beta(f_a, r_f)$ no está definida.

En la figuras 1 se muestran las operaciones posibles λ (registrar), η (reconocer), y β (recuperar) respectivamente.

Computación con tablas La implementación del modo de RIC en el formato de tablas se denomina aquí Computación con Tablas (CT)[5]. Esta representación consiste en una tabla con n columnas y m filas, donde cada una representa un Registro de Memoria Asociativo que contiene una relación entre el conjunto A argumentos $|A| = n$ y el V valores, $|V| = m$. En cada celda de la tabla es posible agregar el valor v_j del argumento a_i se expresa como un 1 en el renglón j de la columna i.

3.2. Memoria asociativa entrópica pesada

En la implementación original de la Memoria Asociativa Entrópica, el contenido de las unidades básicas de memoria o registros estaban encendidos o apagados, por lo tanto todos los objetos almacenados tenían el mismo peso. En [7] se introdujo una extensión de la EAM llamada Memoria Asociativa Entrópica Pesada (W-EAM o Weighted Entropic Associative Memory). Para este modelo se introduce el concepto de pesos en los registros de memoria; en lugar de la disyunción inclusiva, la operación λ (operación Registrar) incrementa en uno el valor de todas las celdas en el registro de memoria asociativa (RMA) correspondiente a las celdas en el registro auxiliar usado por la pista; la operación η (operación Reconocer) se define a través de la implicación material entre las celdas del registro auxiliar y las celdas correspondientes en el RMA; la operación β (operación Recuperar) selecciona una fila del RMA que corresponda al valor del objeto recuperado, para todas las celdas utilizadas por la pista [8]. Estas operaciones se ilustran en la Figura 1.

Tabla 2. Categorías utilizadas en el primer conjunto de experimentos.

Categoría	Etiqueta	Categoría	Etiqueta
Avión	0	Pato	5
Ave	1	Rana	6
Automóvil	2	Caballo	7
Gato	3	Calavera	8
Perro	4	Camión	9

3.3. Arquitectura

Se empleó la CT definida en la sección anterior para implementar un sistema de memoria asociativa entrópica pesada utilizada en el procesamiento de imágenes cuyo modelo se encuentra representado en la Figura 2. Los componentes principales son los mismos que utilizaron en [6]: un encoder (VGG-16-like), un clasificador, y un decoder para la reconstrucción del bosquejo.

4. Propuesta

Para lograr definir los límites que presenta la memoria asociativa entrópica pesada, en este trabajo se optó por trabajar con la base de datos del juego Quick, Draw! creado por Google.

4.1. Quick, Draw!

Quick, Draw! es un juego en línea desarrollado por Google que utiliza la inteligencia artificial para adivinar y reconocer dibujos realizados por el jugador. Véase la figura 3. La base de datos de Quick, Draw! [2] es una colección de 50 millones de imágenes distribuidas en 345 categorías, las cuales han sido contribuidas por los usuarios del juego. Cada imagen es guardada como un conjunto de vectores asociados a una marca de tiempo etiquetados con metadatos los cuales incluyen información como por ejemplo, qué fue lo que se le pidió dibujar al jugador, si el dibujo fue reconocido o no y el país de conexión el jugador. El tamaño de las imágenes fueron de 28×28 píxeles. En la Figura 4 se muestran algunos ejemplos de las imágenes contenidas en ésta base de datos.

5. Experimentación

Para estudiar los parámetros eficientes del uso de la memoria asociativa entrópica pesada se emplearon imágenes de tamaño 28×28 píxeles, y una sola memoria con $\sigma = 0.1$ como en [8]. La distribución de los datos fue de la siguiente manera, 70 % para entrenamiento del codificador y el decodificador, y el clasificador, 20 % para el llenado de la memoria, y el restante 10 % fue para pruebas. El sistema en general se configuró de la siguiente manera:

1. Se entrenan el clasificador y el autoencoder simultáneamente utilizando el conjunto de datos de entrenamiento, del cual se utiliza el 80 % para entrenar las redes y el 20 % para validación.
2. Se prueba el clasificador completo y el autocodificador usando el conjunto de datos de prueba completo.
3. Para todos los registros de memoria con co-dominio 2^m, donde $0 \leq m \leq 10$, se llenan con todo el conjunto de datos escogidos para este fin, incluyendo objetos de todas las clases. Posteriormente se evalúa el rendimiento de la operación de reconocimiento η utilizando cada objeto incluido en el conjunto de datos de prueba como señal, clasificando todos los objetos recuperados y comparando la clase asignada a dicho objeto con la clase asignada a la señal; después se seleccionan los registros de memoria con el mejor número de argumentos.
4. Se prueba el desempeño de la operación de reconocimiento η para los registros seleccionados en el paso anterior con las mejores filas llenándolas con diferentes cantidades del conjunto de datos de llenado, o niveles de entropía.
5. Se evalúa el procedimiento utilizando el método estándar de validación cruzada con $k = 10$.

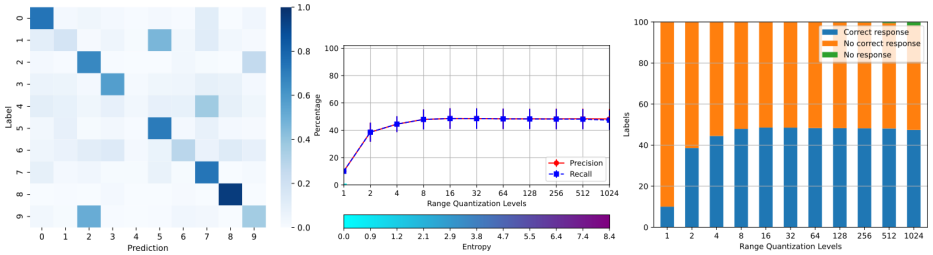
5.1. Tamaño del dominio

Para este conjunto de experimentos se utilizaron 10 categorías (mostradas en la Tabla 2) las cuales contienen 10,000 elementos cada una. El objetivo de estos experimentos es determinar el dominio sobre el cual la memoria presenta un rendimiento óptimo. La Figura 5 muestra los resultados obtenidos para los distintos dominios, se muestran de izquierda a derecha: La matriz de confusión del clasificador; la precisión, recuperación y entropía promedio del reconocimiento del sistema, el dominio indica las columnas (argumentos) de las tablas, el eje horizontal muestra el número de filas y el eje vertical muestra los porcentajes de precisión y recuperación; así como el comportamiento del sistema.

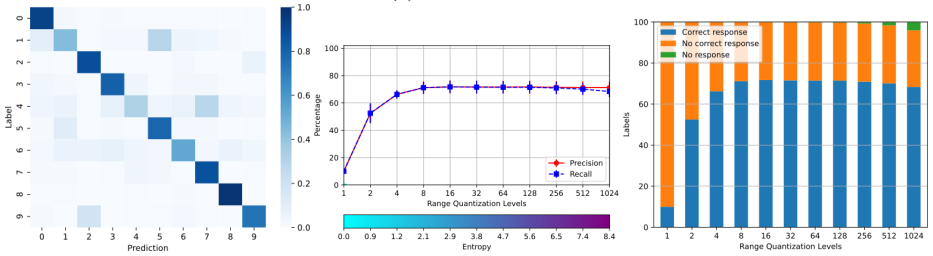
Al analizar los resultados se puede observar que el sistema tiene un mejor rendimiento con 128 columnas, en este caso el clasificador obtuvo una precisión de 80.4 %, de igual manera los porcentajes de precisión y recuperación alcanzaron el 80 %. En la Figura 5 puede observar cómo el porcentaje de precisión y recuepreación se mantiene casi constante al ir aumentando el número de filas, en este caso al estar probando los límites de la memoria se elige la memoria más grande con mejor resultado, la cual sería de tamaño 128×256. Un registro de este tamaño contiene $128 \times 256 = 32,768$ celdas, cada una contiene 2 bytes que guardan números enteros, es decir, esta memoria utiliza 65,536 bytes.

5.2. Número de categorías

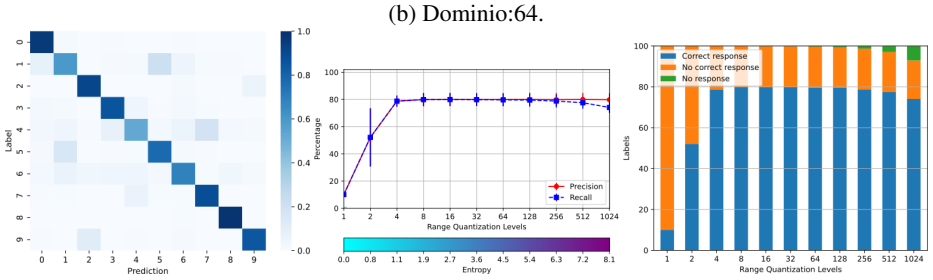
El objetivo de este conjunto de experimentos es determinar con que cantidad de categorías presenta un mejor rendimiento utilizando como base el mejor resultado del primer conjunto de experimentos, es decir, un sistema que presenta una memoria de dominio 128 con 10 categorías con 10,000 imágenes cada una.



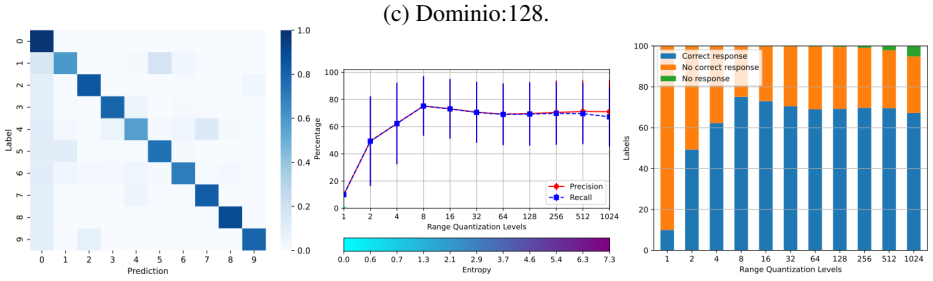
(a) Dominio:32.



(b) Dominio:64.



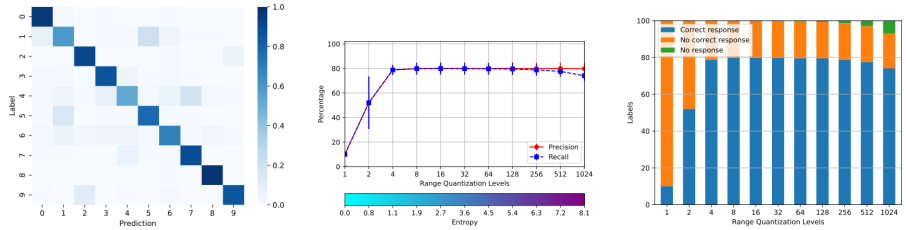
(c) Dominio:128.



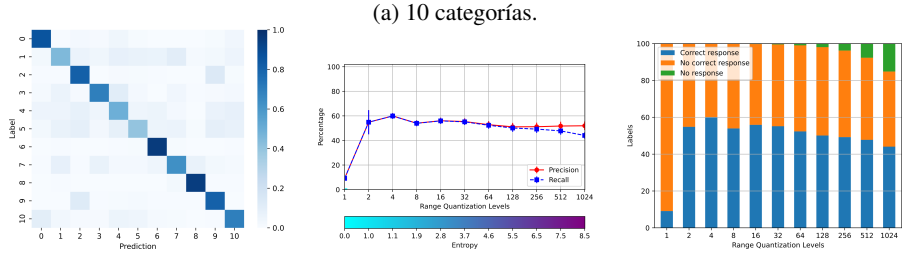
(d) Dominio:256.

Fig. 5. Resultados del experimento con distintos dominios.

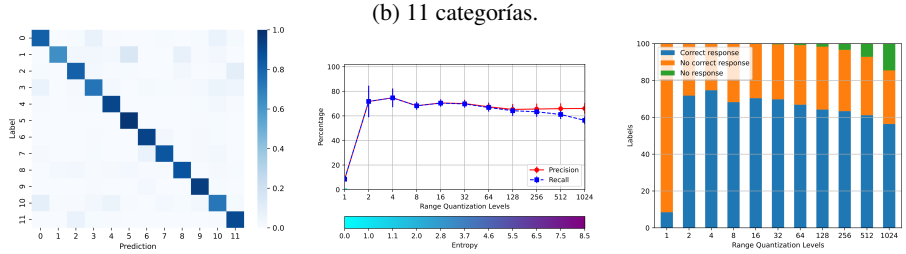
Se utilizan los mismos parámetros así como la misma distribución de los datos mostrados anteriormente. Se realiza el mismo procedimiento que en el conjunto de experimentos anterior pero en esta ocasión variando la cantidad de categorías, utilizando las mostradas en el cuadro 1. En la Figura 6 se presentan los resultados de estos experimentos, se observa como al aumentar la cantidad de categorías a partir de 10 los porcentajes de precisión y recuperación disminuyen al igual que la precisión



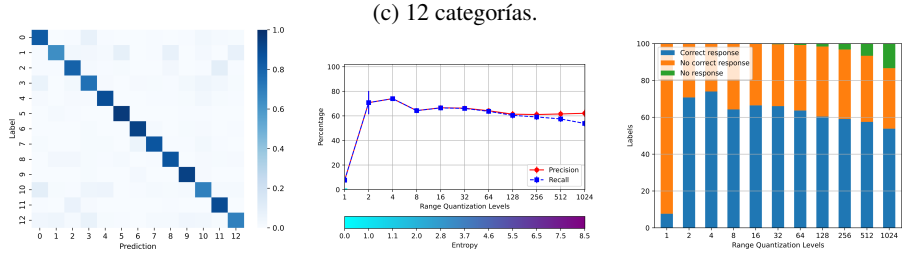
(a) 10 categorías.



(b) 11 categorías.



(c) 12 categorías.



(d) 13 categorías.

Fig. 6. Resultados del experimento con distintos números de categorías.

del clasificador. Sin embargo, los experimentos con 12 y 13 categorías mostraron un buen rendimiento con una memoria de tamaño 128×4 alcanzando 75 % de precisión y recuperación. De acuerdo con los resultados de este conjunto de experimentos la cantidad de categorías con las cuales el sistema presenta un mejor resultado es 10 con 80 % de precisión y recuperación al igual que en el conjunto de experimentos anterior.

6. Conclusiones

Las memorias naturales en humanos y animales son asociativas, declarativas y distribuidas, así que, cuando recordamos algo, las señales o descripciones activan una cadena de recuerdos relacionados basados en significados o contenidos.

De acuerdo con los resultados obtenidos en este trabajo se puede concluir que, dentro de los valores experimentados, la configuración más eficiente y de mayor capacidad de memoria asociativa entrópica pesada para un problema de clasificación de bosquejos de 28×28, es utilizar una memoria de tamaño 128×256, con 10 categorías y 10,000 elementos por categoría obteniendo 80 % de precisión.

Finalmente, una enorme ventaja de la memoria asociativa entrópica es su capacidad de comprimir. En este caso, cada imagen (de tamaño 28×28 píxeles y en formato PNG) extraída de la base de datos en promedio ocupaba 380 bytes; para llenar las memorias se utilizó el 20 % del corpus, es decir 20,000 imágenes que en formato PNG, lo que significa usar 7.6 MB. Al utilizar un sistema de memoria asociativa entrópica pesada se utilizaron 65,536 byte, es decir 0.065536 MB. Por lo tanto el factor de compresión de 116, lo que quiere decir que el espacio ocupado por la memoria distribuida es 115.9 veces menor, y permite tener ahorros de espacio de hasta un 99.1 %.

Debido a la capacidad del sistema de realizar búsquedas en paralelo así como la capacidad de rechazar patrones que no se encuentran alojados en la memoria de manera eficiente, se puede utilizar en sistemas de gestión de bases de datos para recuperar rápidamente datos en función de su contenido ó también se puede utilizar en procesamiento de imágenes. Como trabajo futuro se planea utilizar otros dominios para estudiar el comportamiento de la memoria asociativa entrópica.

Referencias

1. Anderson, J., Bothell, D., Byrne, M., Douglass, S., Lebiere, C., Qin, Y.: An integrated theory of the mind. Psychological review, vol. 111, no. 4, pp. 1036 (2004) doi: 10.1037/0033-295X.111.4.1036
2. Google Creative Lab: The quick, draw! dataset (2017) github.com/googlecreativelab/quickdraw-dataset
3. Google Creative Lab: Quick, draw! (2024) quickdraw.withgoogle.com
4. Morales, R., Hernández, N., Cruz, R., Cruz, V. D., Pineda, L. A.: Entropic associative memory for manuscript symbols. PLOS ONE, vol. 17, no. 8, pp. 1–27 (2022) doi: 10.1371/journal.pone.0272386
5. Pineda, L. A.: Racionalidad Computacional. Academia Mexicana de Computación (2021)
6. Pineda, L. A., Fuentes, G., Morales, R.: An entropic associative memory. Scientific Reports, vol. 11, no. 1, pp. 6948 (2021) doi: 10.1038/s41598-021-86270-7
7. Pineda, L. A., Morales, R.: Weighted entropic associative memory and phonetic learning. Scientific Reports, vol. 12, no. 1, pp. 16703 (2022) doi: 10.1038/s41598-022-20798-0
8. Pineda, L. A., Morales, R.: Imagery in the entropic associative memory. Scientific Reports, vol. 13, no. 1, pp. 9553 (2023) doi: 10.1038/s41598-023-36761-6
9. Quillian, M. R.: Semantic memory. Air Force Cambridge Research Laboratories, Office of Aerospace Research. (1966)
10. Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Gruber, L., Holzleitner, M., Pavlovic, M., Sandve, G. K., Greiff, V., Kreil, D. P., Kopp, M., Klambauer, G., Brandstetter, J., Hochreiter, S.: Hopfield networks is all you need (2020) doi: 10.48550/arXiv.2008.02217
11. Shannon, C. E.: A mathematical theory of communication. The Bell System Technical Journal, vol. 27, pp. 379–423 (1948) doi: 10.1002/j.1538-7305.1948.tb01338.x

ISSN 1870-4069

Modelado 3D de una estructura ósea escaneada con un sensor RGB-D

Ana Valeria Zumaya-García

Centro de Investigaciones en Óptica,
Guanajuato
México

anavaleria@cio.mx

Resumen. En este artículo, se presenta una metodología para generar el modelo 3D de una estructura ósea a partir de la nube de puntos obtenida mediante el escaneo 3D utilizando un sensor de profundidad (RGB-D), específicamente el sensor Kinect V2.0. Se aplicaron tres métodos de modelado 3D: Greedy, Poisson y Grid Projection, permitiendo una visualización tridimensional de las dimensiones reales de la estructura ósea. Se comprobó que el método Greedy posee las dimensiones más cercanas a la estructura ósea real, debido a que los métodos Poisson y Grid Projection presentaron mayores variaciones de acuerdo con las dimensiones reales de la estructura ósea.

Palabras clave: Kinect V2.0, modelo 3D, estructura ósea, escaneo, nube de puntos.

3D Modeling of a Bone Structure Scanned with an RGB-D Sensor

Abstract. This article presents a methodology for generating the 3D model of a bone structure from the point cloud obtained through 3D scanning using a depth sensor (RGB-D), specifically the Kinect V2.0 sensor. Three 3D modeling methods were applied: Greedy, Poisson, and Grid Projection, allowing a three-dimensional visualization of the actual dimensions of the bone structure. It was found that the Greedy method has the closer dimensions of the actual bone structure. In contrast, the Poisson and Grid Projection methods showed more significant variations according to the actual dimensions of the bone structure.

Keywords: Kinect V2.0, 3D model, bone structure, scanning, point cloud.

1. Introducción

En el contexto actual, donde la tecnología y la inteligencia artificial (IA) están transformando múltiples sectores, el diseño asistido por computadora (CAD) está experimentando un cambio significativo. Los modelos de objetos, utilizados en campos como la arquitectura, medicina, industria, educación y artes, están evolucionando gracias a la integración de métodos de IA [11].

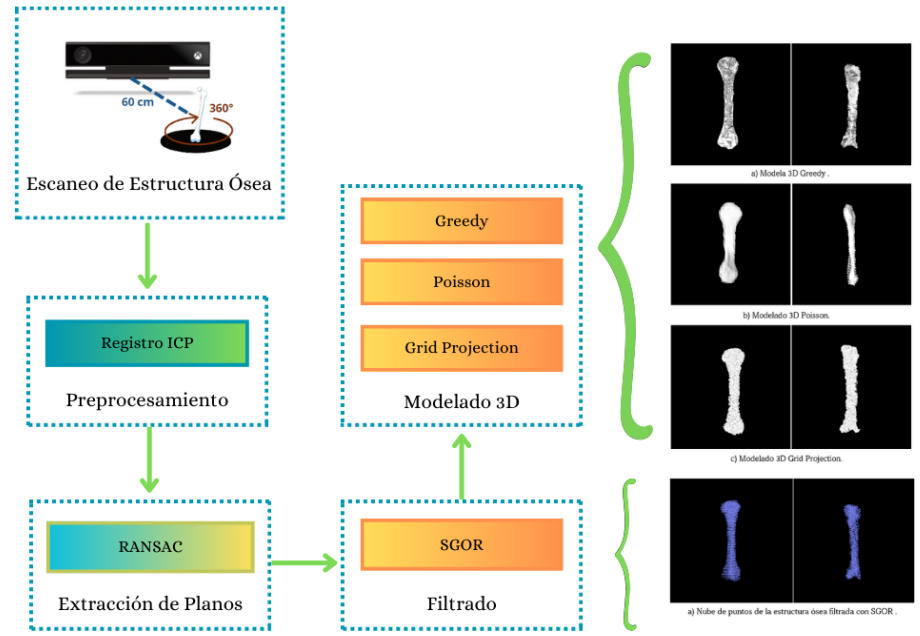


Fig. 1. Diagrama que ilustra las cinco etapas de la metodología propuesta para obtener un modelo 3D a partir de una estructura ósea escaneada con el sensor Kinect V2.0, utilizando tres métodos de modelado 3D: Greedy, Poisson y Grid Projection. Las etapas son: escaneo, captura de la estructura ósea desde múltiples ángulos; registro, alineación de las capturas parciales para formar una vista global coherente de la estructura ósea real; extracción de planos, identificación y eliminación de valores atípicos en la nube de puntos de la estructura ósea; filtrado, limpieza de la nube de puntos para mejorar la precisión; modelado 3D, aplicación de tres métodos de modelado para crear modelos tridimensionales: Greedy, Poisson y Grid Projection.

Este artículo se enfoca en comparar modelos digitales tridimensionales de una estructura ósea real, preservando sus dimensiones reales para permitir su adaptación, modificación y reconstrucción. Para lograr este objetivo, se aplicaron técnicas de reconocimiento de patrones, garantizando una mayor precisión y propiedades mejoradas. La digitalización simplificada reducirá los tiempos y costos asociados al diseño. Es fundamental que los métodos utilizados sean escalables, permitiendo modelar no solo estructuras óseas generales, sino también objetos específicos. En esta etapa inicial, se busca mejorar y validar los métodos aplicados.

La sinergia entre el reconocimiento de patrones y la inteligencia artificial promete revolucionar la manera en que creamos y transformamos nuestro entorno construido. En este sentido, el escaneo de la estructura ósea se realizará con el sensor Kinect V2.0; generando una nube de puntos, junto con el método de filtrado SGOR y los métodos de modelados 3D: Greedy, Poisson y Grid Projection, basados en el estado del arte, como se ha demostrado en los artículos de los autores Kang [4], Juszczyk [3] y Enguo [1]. Se destaca el método de filtrado SGOR [6] y los métodos de modelado Greedy [5], Poisson [2] y Grid Projection [13], que proporcionan una base sólida para el desarrollo de nuestra metodología.

(a) Frente de la estructura ósea (b) Lado Frontal de la estructura ósea

Fig. 2. Estructura Ósea Real; en (a) se observa la estructura ósea con sus dimensiones de frente, mientras que en (b) se muestra la estructura ósea con sus dimensiones del lado frontal. Esta estructura ósea fue seleccionada con el fin de analizar sus dimensiones físicas en las diferentes etapas de la metodología propuesta.

Este artículo, se centra en los trabajos realizados por Syed [12] y Mahmood [9], presentan métodos de modelado similares a los que se proponen, permitiendo evaluar la eficacia y las ventajas de nuestra metodología en términos de precisión y fidelidad en la representación de la estructura ósea digital. La principal contribución de este artículo es identificar el mejor modelo tridimensional de los diferentes métodos de modelado 3D: Greedy, Poisson y Grid Projection. Para esto fue necesario implementar y optimizar los algoritmos que se describen en la sección 2. El escrito se divide en cuatro secciones metodología 2, resultados 3, conclusiones 4 y trabajo a futuro 5.

2. Metodología

El planteamiento del problema esta enfocado en obtener un modelo 3D a partir de una estructura ósea escaneada con el sensor Kinect V2.0, utilizando tres métodos de modelado 3D: Greedy, Poisson y Grid Projection. La metodología propuesta en este artículo, se divide en cinco etapas principales: escaneo, registro, extracción de planos, filtrado y modelado 3D. En la primera etapa, se escanea la estructura ósea desde múltiples ángulos utilizando una platina giratoria para obtener una visión completa de 360° de la estructura ósea, lo que agiliza la adquisición de datos y reduce la necesidad de mano de obra especializada.

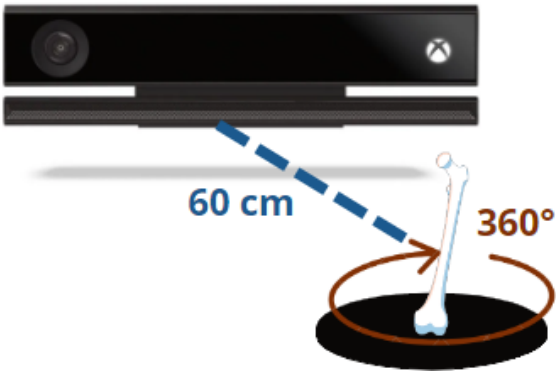
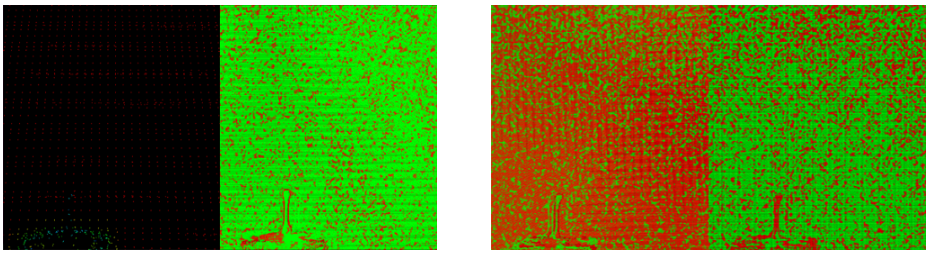


Fig. 3. Entorno de escaneo para la adquisición de la estructura ósea.



(a) Inicio del proceso del registro (b) Fin del proceso del registro

Fig. 4. Proceso de registro de los 16 escaneos en intervalos de 22.5° de las nubes de puntos de la estructura ósea, para obtener una sola nube de puntos de la estructura ósea en conjunto de todas sus vistas.

En la segunda etapa, se utiliza la técnica de registro punto más cercano iterativo (ICP, por sus siglas en inglés) [8], para alinear y fusionar las múltiples capturas parciales en una vista global coherente de la estructura ósea. Posteriormente, en la tercera etapa, se emplea el método de consenso de muestra aleatoria (RANSAC, por sus siglas en inglés) [7], para identificar y eliminar valores atípicos, mejorando así la eficiencia de las siguientes etapas. En la cuarta etapa, se aplica el filtro de eliminación estadística bruta de valores atípicos (SGOR, por sus siglas en inglés) para limpiar la nube de puntos de la estructura ósea de datos no deseados o ruidosos, asegurando la precisión del modelado 3D.

Finalmente, en la quinta etapa, se utilizan tres métodos de modelado 3D: Greedy, Poisson y Grid Projection. Greedy opera de manera iterativa para construir una malla 3D eficiente, mientras que Poisson utiliza campos de gradientes para una reconstrucción suave y precisa. Por otro lado, Grid Projection divide el espacio en una cuadrícula tridimensional para generar triángulos que representan la superficie de la estructura ósea, siendo una opción robusta para datos de baja resolución. Estos métodos ofrecen diferentes enfoques para crear modelos 3D cercanos a las dimensiones de la estructura ósea escaneada. La Fig. 1 proporciona una guía visual para comprender la metodología propuesta.

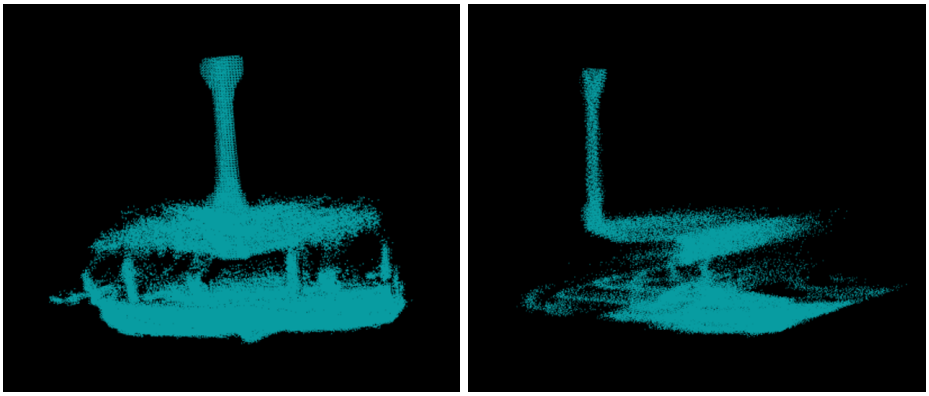
Fig. 5. Nube de puntos final de la estructura ósea obtenida en la etapa de registro ICP se alinea con los 16 escaneos en intervalos de 22.5°, mostrando una sola nube de puntos que engloba a los múltiples escaneo.

3. Resultados

En esta sección, se presentan los resultados obtenidos después de aplicar las técnicas y los métodos descritos en la metodología de la sección 2. Se muestran las imágenes del registro, la extracción de planos y el resultado del filtrado. Finalmente, se muestran los resultados obtenidos de los tres métodos de modelado 3D: Greedy, Poisson y Grid Projection.

3.1. Escaneo de estructura ósea

Para el escaneo se utilizó una estructura ósea correspondiente al fémur de un pollo; en la Fig. 2 se muestran las dimensiones de la estructura ósea real. En la Fig. 2a se observa la estructura ósea con sus dimensiones de frente, mientras que en la Fig. 2b se muestra la estructura ósea con sus dimensiones del lado frontal. Esta estructura ósea fue seleccionada con el fin de analizar sus dimensiones físicas en las diferentes etapas de la metodología. Posteriormente se utilizó el sensor Kinect V2.0 en conjunto con la platina giratoria. El sensor Kinect V2.0 captura 16 vistas en intervalos de 22.5° para obtener la información completa de los 360° de la estructura ósea. Esta información se presenta en forma de nubes de puntos, cada una correspondiente a una de las 16 vistas capturadas con el sensor Kinect V2.0. La platina giratoria está equipada con un motor a pasos para controlar sus movimientos, lo que permite obtener información de los 360° de la estructura ósea y facilita la adquisición de múltiples vistas parciales.



(a) Frente de la nube de puntos (b) Lado frontal de la nube de puntos

Fig. 6. Resultado de la extracción centrándose principalmente en el plano pared de la nube de puntos de la Fig. 5. En a), se observa la estructura ósea con el plano extraído de frente, mientras que en b), se muestra la estructura ósea con el plano extraído del lado frontal.

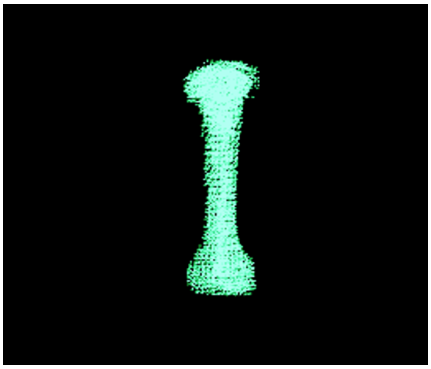
Para llevar a cabo el escaneo, la estructura ósea se coloca sobre la platina giratoria, cuyo centro se encuentra a 60 centímetros del sensor Kinect V2.0, como se muestra en la Fig. 3. Este parámetro permite obtener la mayor cantidad de dimensiones de la estructura ósea, así como disminuir el ruido durante el escaneo. El proceso de escaneo se analizó de esta manera para garantizar la obtención de escaneos completos y detallados de la estructura ósea, lo que es fundamental para su posterior análisis y pre-procesamiento.

3.2. Registro

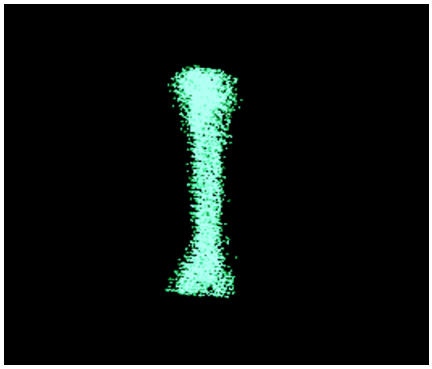
Una vez que se han escaneado los 360° de la estructura ósea, se aplica la técnica de registro. Para ello, se configuró la librería de nubes de puntos (PCL, por sus siglas en inglés) [10], y se implementó la técnica ICP. El proceso consiste en combinar los diferentes ángulos de visión de los escaneos obtenidos en la etapa anterior, superponiendo las dimensiones correspondientes de las distintas nubes de puntos.

El proceso de registro se puede observar en la Fig. 4; esto permite visualizar todas las nubes de puntos como un conjunto global que posee el modelo con todas sus vistas. En la Fig. 4a se muestra el inicio del proceso de registro de la estructura ósea, donde del lado derecho se aprecian las nubes de puntos obtenidas desde diferentes ángulos de escaneo, mientras que del lado izquierdo se muestra una representación visual del proceso de registro.

Este último puede incluir líneas de conexión entre puntos coincidentes en las diferentes vistas, que van desapareciendo a medida que avanza el proceso. Por otro lado, la Fig. 4b ilustra el fin del proceso de registro de la estructura ósea, donde del lado derecho se observa la fusión completa de todas las nubes de puntos en una sola representación tridimensional, mientras que del lado izquierdo se pueden mostrar métricas de calidad del registro, como la cantidad de puntos coincidentes o la distancia entre las nubes de puntos registradas.

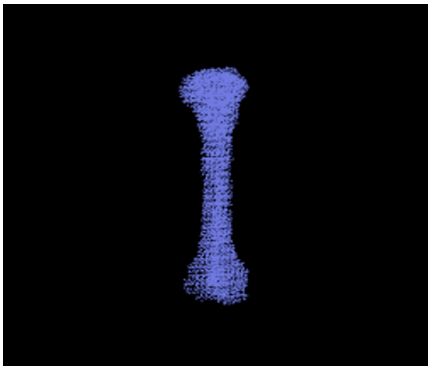


(a) Frente de la estructura ósea 3D

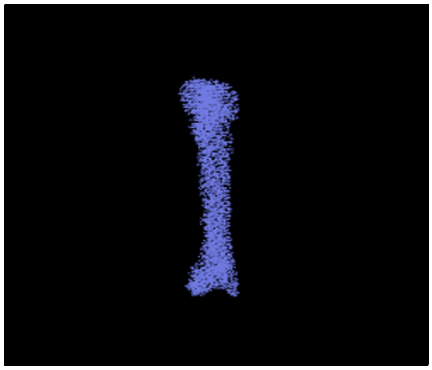


(b) Lado frontal de la estructura ósea 3D

Fig. 7. Resultado de la extracción del plano centrándose principalmente en la platina giratoria de la nube de puntos de la Fig. 6. En a), se observa la estructura ósea 3D con sus dimensiones de frente luego de extraer la platina, mientras que en b), se muestra la estructura ósea con sus dimensiones del lado frontal después de extraer la platina.



(a) Frente de la estructura ósea 3D con filtro SGOR



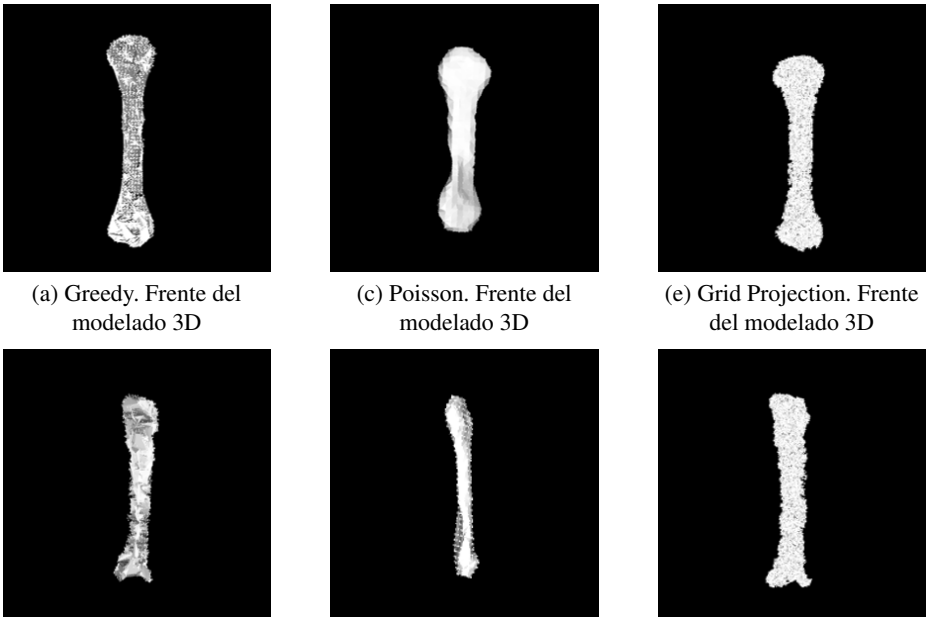
(b) Lado frontal de la estructura ósea 3D con filtro SGOR

Fig. 8. Resultado al aplicar el filtro SGOR en la Fig. 7. En a), se observa la estructura ósea 3D con sus dimensiones de frente luego de aplicar el filtro SGOR, mientras que en b), se muestra la estructura ósea con sus dimensiones del lado frontal después de aplicar el filtro SGOR.

Como resultado del proceso de registro de las nubes de puntos, se obtiene una sola nube dme puntos que representa la forma y geometría de la estructura ósea; en la Fig. 5 se muestra el resultado final del registro, donde se puede apreciar la estructura ósea completa y alineada en una sola nube de puntos.

3.3. Extracción de planos

Después de obtener el registro de las nubes de puntos de la estructura ósea, es necesario extraer o eliminar la información irrelevante. Se utilizó el método RANSAC para identificar y extraer la información irrelevante generada durante los escaneos, con el fin de extraer el plano y la platina giratoria de la Fig. 5.



(a) Greedy. Frente del modelado 3D

(c) Poisson. Frente del modelado 3D

(e) Grid Projection. Frente del modelado 3D

(b) Greedy. Lado frontal del modelado 3D

(d) Poisson. Lado frontal del modelado 3D

(e) Grid Projection. Lado frontal del modelado 3D

Fig. 9. Resultado cualitativo al aplicar los tres métodos de modelado 3D: Greedy, Poisson y grid projection. En a) y b), se muestra el resultado del modelado 3D de la estructura ósea con Greedy. En c) y d) se observa el resultado de la técnica de Poisson. Finalmente, en e) y f) se muestra el resultado de la técnica de grid projection.

En la Fig. 6 se observa el resultado de extracción de un plano, centrándose principalmente en el plano de la pared; en la Fig. 6a, se observa la estructura ósea con el plano extraído de frente, mientras que en la Fig. 6b, se muestra la estructura ósea con el plano extraído del lado frontal. Una vez extraído el plano, el método identificó los puntos que mejor se ajustan al plano de la platina, excluyendo aquellos que no forman parte de esta estructura ósea. Esto garantiza que se conserven únicamente los puntos relevantes para el análisis y la reconstrucción de la estructura ósea, evitando la inclusión de ruido o puntos que no forman parte de la superficie de interés. Finalmente en la Fig. 7 se observa la estructura ósea 3D, después de extraer la platina giratoria; en la Fig. 7a, se observa la estructura ósea 3D con sus dimensiones de frente luego de extraer la platina, mientras que en la Fig. 7b, se muestra la estructura ósea con sus dimensiones del lado frontal después de extraer la platina.

3.4. Filtrado

En esta etapa, se busca eliminar los datos atípicos de la estructura ósea 3D. Se empleó el filtro estadístico SGOR para determinar los puntos que debían ser eliminados; este tipo de filtro es especialmente útil, ya que los dispositivos que capturan este tipo de estructuras tienden a generar nubes de puntos muy ruidosas que dificultan la distinción de ciertas dimensiones del modelo.

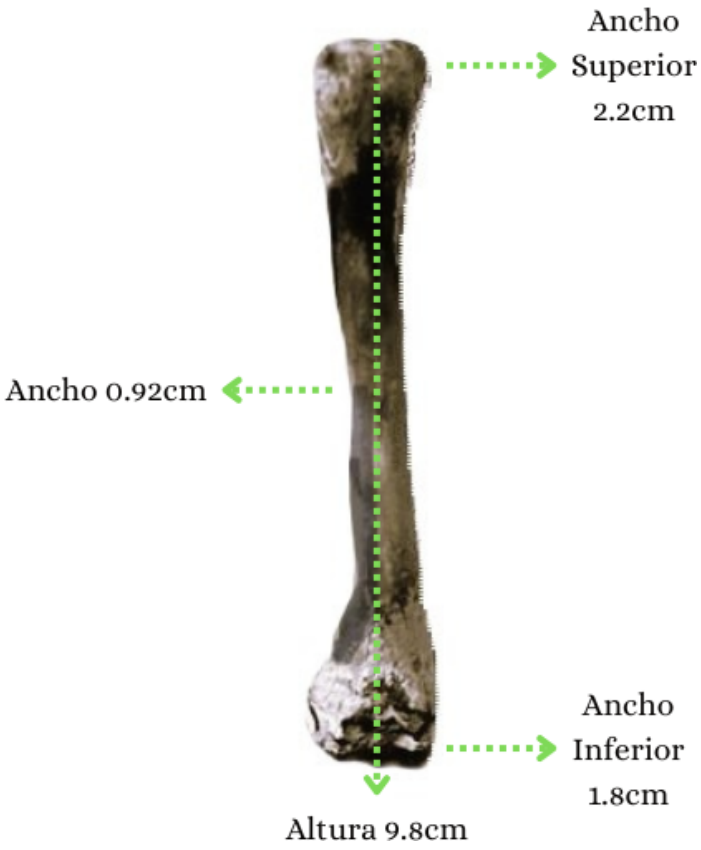


Fig. 10. Dimensiones reales de la estructura ósea.

Para verificar la efectividad del método, se utilizó la nube de puntos de la estructura ósea 3D mostrada en la Fig. 7. En la Fig. 8, se muestra el resultado obtenido al aplicar el filtro SGOR sobre la nube de puntos de la estructura ósea 3D de la Fig. 7. En esta imagen, se pueden apreciar con mayor claridad las dimensiones de la estructura ósea real, como se muestra en la Fig. 2. Se observa la notable eliminación de puntos atípicos y ruido presentes en la nube de puntos, generados por el sensor Kinect V2.0.

3.5. Modelado 3D

El proceso de mallado o modelado de la nube de puntos desempeña un papel crucial al permitir la reproducción digital de las dimensiones de la estructura ósea escaneada. En esta etapa, se presentan los resultados obtenidos de la aplicación de los métodos de modelado 3D: Greedy, Poisson y Grid Projection. Cada uno de estos métodos emplea enfoques distintos para generar superficies tridimensionales a partir de la nube de puntos filtrada de la estructura ósea. Los resultados de estos métodos proporcionan una representación visual de las dimensiones precisas de la estructura ósea escaneada, lo que facilita su análisis y visualización en entornos digitales.

Tabla 1. Análisis de las medidas utilizando el software MeshLab, con el fin de evaluar la eficacia de los tres métodos de modelado 3D: Greedy, Poisson y grid projection.

Dimensiones de la estructura ósea	Medidas reales de la estructura ósea	Medidas del modelado 3D Greedy	Medidas del modelado 3D Poisson	Medidas del modelado 3D grid
Altura	9.8 cm	9.6 cm	10 cm	9.4 cm
Ancho	0.92 cm	0.9 cm	1.3 cm	0.95 cm
Ancho Superior	2.2 cm	2.2 cm	2. 8cm	2.3 cm
Ancho Inferior	1.8 cm	2.0 cm	2.5 cm	2.2 cm

Tabla 2. Errores Absolutos para los métodos de modelado 3D: Greedy, Poisson y grid projection.

Dimensiones de la estructura ósea	Error absoluto modelado 3D Greedy	Error absoluto modelado 3D Poisson	Error absoluto modelado 3D grid
Altura	0.2 cm	0.2 cm	0.4 cm
Ancho	0.02 cm	0.38 cm	0.03 cm
Ancho Superior	0 cm	0.6 cm	0.1 cm
Ancho Inferior	0.2 cm	0.7 cm	0.4 cm

En la Fig. 2 se presenta la imagen de la estructura ósea real, la cual fue seleccionada con el propósito de analizar sus dimensiones físicas, como se mencionó en la sección 3.1. Posteriormente, en la Fig. 9, se muestran los resultados cualitativos del modelado 3D de la estructura ósea tras aplicar los métodos de Greedy, Poisson y Grid Projection. En la Fig. 9a y 9b se puede apreciar el resultado del modelado 3D de la estructura ósea utilizando el método Greedy. En este caso, la superficie original se conserva, permitiendo una modelación adecuada de la forma compleja de la estructura ósea en 3D. En la Fig. 9c y 9d se observa el resultado del método Poisson.

Aunque esta técnica puede producir un modelo suave, tiende a perder información de la superficie original debido a la simplificación excesiva de los detalles. Finalmente, en la Fig. 9e y 9f se muestra el resultado del método Grid Projection. Esta técnica permite obtener la superficie original de manera efectiva, brindando una representación visual detallada de la estructura ósea en 3D. Posteriormente, se realizó un análisis utilizando el software MeshLab para evaluar la eficacia de los tres métodos de modelado 3D: Greedy, Poisson y Grid Projection. Se enfocó en comparar las dimensiones reales de la estructura ósea escaneada, centrándose en su altura y ancho incluyendo las medidas superior e inferior.

La Fig. 10 muestra las dimensiones reales de la estructura ósea. Los resultados de la Tabla 1 indican diferencias significativas en las medidas obtenidas mediante los tres métodos de modelado 3D: Greedy, Poisson y Grid Projection. En el análisis, se destaca que el método Greedy puede ofrecer una mayor aproximación en relación con las dimensiones reales de la estructura ósea. Para confirmar esta observación, se llevó a cabo un análisis cuantitativo que incluyó el cálculo del error absoluto (EA) y relativo (ER); estos errores se calcularon utilizando las ecuaciones 1 y 2.

Tabla 3. Errores relativos para los métodos de modelado 3D: Greedy, Poisson y grid projection.

Dimensiones de la estructura ósea	Error relativo modelado 3D Greedy	Error relativo modelado 3D Poisson	Error relativo modelado 3D Grid
Altura	2.04 %	2.04 %	4.08 %
Ancho	2.17 %	41.30 %	3.26 %
Ancho Superior	0 %	27.27 %	4.54 %
Ancho Inferior	11.11 %	38.88 %	22.22 %

El error absoluto proporcionó una medida directa de la variación entre las medidas obtenidas y las dimensiones reales, mientras que el error relativo permitió comparar la aproximación de los métodos independientemente de la escala de las dimensiones:

$$EA = |V_{\text{real}} - V_{\text{med}}|\,, \tag{1}$$

$$ER(\,\%) = \left\| \frac{EA}{V_{\text{real}}} \right\| \times 100, \tag{2}$$

donde:

– V_{real}, es la medida real de las dimensiones de la estructura ósea.

– V_{med}, es la medida modelada de las dimensiones de los métodos de modelado 3D: Greedy, Poisson y Grid Projection.

Al analizar los resultados, se observa que los errores proporcionan una medida directa de la variación entre las medidas calculadas y las dimensiones reales en unidades de centímetros (ver Tabla 2). Se mostró que el método de modelado 3D Greedy tiene una menor variación en todas las dimensiones de la estructura ósea en comparación con las dimensiones reales. En la Tabla 3 los errores relativos expresan la variación porcentual entre las medidas calculadas y las dimensiones reales, proporcionando una comparación de la aproximación relativa de cada método de modelado 3D. Estos resultados comprobaron que el método Greedy proporciona una representación más aproximada de las dimensiones reales de la estructura ósea escaneada en comparación con los métodos Poisson y Grid Projection, además de tener un procesamiento más rápido.

4. Conclusiones

Se llevó a cabo un estudio de optimización de los métodos de modelado de datos 3D obtenidos a partir de la estructura ósea escaneada utilizando el sensor Kinect V2.0; este estudio ha demostrado que el sensor Kinect V2.0 proporciona buena calidad de información durante el escaneo de la estructura ósea, que junto con las técnicas de pre-procesamiento de datos propuestas, permite obtener modelos 3D precisos y detallados. Se logró obtener los resultados esperados de la técnica de registro ICP y de la extracción de planos con RANSAC en la nube de puntos de la estructura ósea.

Además, la técnica de filtrado SGOR demostró ser altamente efectiva, proporcionando una mejora significativa en la calidad de la estructura ósea 3D; por la eliminación de los puntos atípicos y el ruido presente en la nube de puntos, derivado del escaneo con el sensor Kinect V2.0. Se observó que el proceso de eliminación de datos atípicos puede repetirse varias veces para lograr un mejor filtrado, optimizando así los datos obtenidos por el sensor Kinect V2.0. Durante la etapa de modelado, se aplicaron los métodos Greedy, Poisson y Grid Projection. Los resultados obtenidos se compararon según las dimensiones de la estructura ósea real. Se observó que Greedy sobresale por su capacidad para preservar detalles finos y la geometría original de la superficie, especialmente en estructuras complejas. Aunque Grid Projection puede ser más eficiente computacionalmente y adecuado para superficies más simples, Greedy ofrece una mejor fidelidad en la reproducción de la forma original, lo que lo convierte en la elección adecuada cuando se prioriza la precisión y el detalle del modelado tridimensional.

5. Trabajo a futuro

Como trabajo a futuro, se realizarán pruebas adicionales utilizando el método de modelado 3D Greedy en la impresión 3D. Se contemplará una variedad de estructuras óseas y objetos de instrumentación médica. Se espera que esta metodología tenga un impacto significativo en dos áreas clave: la educación médica, al mejorar la enseñanza de anatomía, y la manufactura aditiva, al facilitar la creación de implantes personalizados y prótesis utilizando materiales biocompatibles.

Referencias

1. Enguo, W., Mengxiang, C., Chengzhi, S., Xiaochen, Z.: Research on point cloud fine registration method combined with colored grid projection. In: Proceedings of the 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, pp. 394–399 (2022) doi: 10.1109/cyber55403.2022.9907649
2. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Surface reconstruction from unorganized points. In: Proceedings of the 19th annual conference on Computer graphics and interactive techniques, pp. 71–78 (1992) doi: 10.1145/133994.134011
3. Juszczyk, J. M., Wijata, A., Czajkowska, J., Krecichwost, M., Rudzki, M., Biesok, M., Pycinski, B., Majewski, J., Kostecki, J., Pietka, E.: Wound 3D geometrical feature estimation using poisson reconstruction. IEEE Access, vol. 9, pp. 7894–7907 (2021) doi: 10.1109/access.2020.3035125
4. Kang, J., Lee, S.: A greedy pursuit approach for fitting 3D facial expression models. IEEE Access, vol. 8, pp. 192682–192692 (2020) doi: 10.1109/access.2020.3029065
5. PCL: Greedy projection tutorial. PCL Documentation (2021) pcl.readthedocs.io/projects/tutorials/en/latest/greedyprojection.html
6. PCL: Statistical outlier removal tutorial. PCL Documentation (2021) pcl.readthedocs.io/projects/tutorials/en/latest/statisticaloutlier.html
7. Qian, X., Ye, C.: NCC-RANSAC: A fast plane extraction method for 3-D range data segmentation. IEEE Transactions on Cybernetics, vol. 44, no. 12, pp. 2771–2783 (2014) doi: 10.1109/tcyb.2014.2316282

8. Qin, L., Chen, X., Gong, X.: An improved 3D reconstruction method for weak texture objects combined with calibration and ICP registration. In: IEEE 6th International Conference on Industrial Cyber-Physical Systems, vol. 31, pp. 1–5 (2023) doi: 10.1109/icps58381.2023.10128042
9. Riedle, H., Seitz, V., Schraudolf, L., Franke, J.: Generation of 3D silicone models of anatomic soft tissue structures - A comparison of direct 3D printing and molding techniques. In: IEEE-EMBS Conference on Biomedical Engineering and Sciences, vol. 37, pp. 539–543 (2018) doi: 10.1109/iecbes.2018.8626687
10. Rusu, R. B., Cousins, S.: 3D is here: Point cloud library. In: IEEE International Conference on Robotics and Automation (2011) doi: 10.1109/icra.2011.5980567
11. Sabbella, D. S., Singh, A., Maheswari G. U.: Conferencia Internacional sobre Tendencias Emergentes en Tecnología e Ingeniería de la Información. Inteligencia Artificial en Modelado CAD 3D, pp. 1–5 (2020)
12. Syed, H. H., Mahmood, M. H.: 3D human reconstruction with corresponding 3D texture model: A comparison of salient approaches. In: International Conference on Emerging Technologies in Electronics, Computing and Communication, vol. 33, pp. 1–6 (2022) doi: 10.1109/icetecc56662.2022.10068940
13. Wang, W., Yan, C., Huang, Y., Wang, S.: Enhancement of viscous grid projection algorithm and application. In: 7th International Conference on Mechanical and Aerospace Engineering, vol. 38, pp. 578–581 (2016) doi: 10.1109/icmae.2016.7549606

ISSN 1870-4069

Identificación de ángulos con puntos de referencia del cuerpo humano mediante machine learning para personas geriátricas

Mariana Martinez-Hernandez, Benjamín Arturo Perez-Pelaez, Roberto Ángel Melendez-Armenta, David Lara-Alabazares, Irahan-Otoniel José-Guzmán

Tecnológico Nacional de México,
Campus Misantla,
División de Estudios de Posgrado e Investigación,
México

{232t0534, 232t0535, ramelendeza, dlaraa, iojoseg}@itsm.edu.mx

Resumen. En México, la población de adultos mayores está en constante crecimiento debido al aumento de la esperanza de vida y la disminución de las tasas de fecundidad, lo que plantea desafíos sociales y de salud significativos para este grupo demográfico. La investigación en rehabilitación geriátrica ha sido fundamental para comprender el funcionamiento diario de este sector de la población, explorando herramientas y métodos en unidades geriátricas y comparando diferentes enfoques de rehabilitación. Este artículo propone una innovadora forma de apoyo para la salud de los adultos mayores: el procesamiento inteligente de movimientos para la profilaxis geriátrica. Utilizando técnicas como el machine learning, redes neuronales convolucionales y funciones trigonométricas, se busca medir los ángulos sobre los puntos de referencia del cuerpo humano. Este enfoque promete mejorar la calidad de vida de los adultos mayores al detectar y prevenir problemas de movilidad y postura, representando un avance importante en el cuidado geriátrico.

Palabras clave: Machine learning, geriátrica, puntos de referencia, ángulos.

Identification of Angles with Reference Points of the Human Body Using Machine Learning for Geriatric People

Abstract. In Mexico, the elderly population is constantly growing due to increasing life expectancy and decreasing fertility rates, posing significant social and health challenges for this demographic group. Research in geriatric rehabilitation has been fundamental to understand the daily functioning of this sector of the population, exploring tools and methods in geriatric units and comparing different rehabilitation approaches. This article proposes an innovative way to support the health of older adults: intelligent movement processing for geriatric prophylaxis. Using techniques such as machine learning, convolutional neural networks and trigonometric functions, the aim is to measure the angles on

the reference points of the human body. This approach promises to improve the quality of life of older adults by detecting and preventing mobility and posture problems, representing an important advance in geriatric care.

Keywords: Machine learning, geriatric, landmarks of the human body, angles.

1. Introducción

De acuerdo a las investigaciones dirigidas por el doctor Sergio Salvador Valdés y Rojas, director de Atención Geriátrica del INAPAM, el sedentarismo se posiciona como un asunto de preocupación primordial en el contexto de la sociedad contemporánea. En este sentido, se destaca la imperiosa necesidad de promover entre las diversas cohortes generacionales la práctica regular de actividad física, especialmente aquella de naturaleza deportiva, como estrategia fundamental para estimular un envejecimiento caracterizado por la vitalidad y la salud óptima.

La función física emerge como un componente esencial en este proceso, dado su papel determinante en la preservación de la capacidad funcional a lo largo del ciclo vital. Desde la perspectiva del adulto mayor, la inclusión sistemática de actividad física en la rutina diaria se erige como una de las intervenciones más trascendentales para salvaguardar el bienestar general. Este enfoque no solo se proyecta como un medio efectivo para prevenir o mitigar la incidencia de diversas condiciones de salud vinculadas con el proceso de envejecimiento, sino que también se reconoce por su capacidad para potenciar la musculatura, facultando así la independencia en la ejecución de las actividades cotidianas, sin requerir asistencia externa [12].

En México, el último dato es que habitan casi 13 millones de personas con una edad de 60 años o más. El aumento de la esperanza de vida junto con la disminución de las tasas de fecundidad ha dado como resultado un progresivo envejecimiento de la población, lo que conlleva retos sociales y de salud relativos a este grupo etario [10]. En este contexto, es crucial considerar la problemática derivada del aumento en la cantidad de personas mayores que necesitarán atención médica especializada en rehabilitación geriátrica.

Dada la limitada capacidad del personal en geriatría para satisfacer esta creciente demanda, resulta imperativo buscar soluciones innovadoras. Por consiguiente, este estudio propone una metodología que a través del procesamiento inteligente de ángulos en puntos de referencia del cuerpo humano, permitiendo desarrollar sistemas inteligentes para la rehabilitación geriátrica en las personas de edad avanzada.

2. Marco teórico

La rehabilitación geriátrica se ha beneficiado de múltiples estudios como referencia para conocer el desempeño en actividades físicas básicas de adultos mayores, dentro de la investigación se conocieron herramientas y métodos en estancias de unidades geriátricas, así como la comparación de los resultados entre distintos enfoques de rehabilitación.

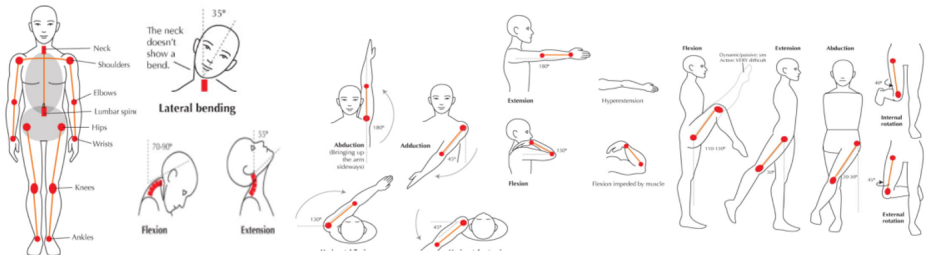


Fig. 1. Puntos de referencia del cuerpo humano.

2.1. Estado de arte

Carmona, et. al, Observaron que el envejecimiento poblacional es un fenómeno. Se espera que el número de personas mayores de 60 años aumente más del doble en 2050 y del triple en 2100, pasando de 962 millones en todo el mundo en 2017 a 2.100 millones en 2050 y 3.100 millones en 100. Inevitable, siendo un reto para el sistema de salud de la comunidad mundial, utilizar la Bioética como método de reflexión, la aplicación correcta de los principios bioéticos en rehabilitación geriátrica por parte de los profesionales de salud se contribuye a elevar la calidad de vida y de la asistencia médica de este grupo poblacional [7].

Baztan, et. al, Describir la evolución y resultados de la rehabilitación de ancianos incapacitados atendidos en una unidad geriátrica de media estancia y conocer los factores asociados a mejoría funcional e institucionalización al alta. Se estudiaron todos los pacientes ingresados consecutivamente en la Unidad de Media Estancia del servicio de Geriatría del Hospital Central de la Cruz Roja de Madrid desde mayo de 2000 hasta diciembre de 2001. se utilizó la regresión logística introduciendo todas las variables consideradas en el análisis bivariante.

Los datos se analizaron en el paquete estadístico SPSS 9.0, Se ingresaron 506 pacientes en la unidad en el período de estudio, de los que se excluyeron 47 para el análisis (31 derivados al alta a unidades de agudos, 7 permanecieron menos de 5 días en la unidad, 5 fallecieron y 4 por datos insuficientes) [1]. Llosa and Gutiérrez, explicaron la utilización de la música en la rehabilitación geriátrica para la formación del Licenciado en Rehabilitación en Salud, cuenta con un Programa Nacional Integral al Adulto Mayor, el método utilizado fue la música en la rehabilitación geriátrica, la música es una terapia no farmacológica no invasiva para las edades avanzadas sanas y enfermas, en la rehabilitación geriátrica para la formación del Licenciado en Rehabilitación en Salud [13].

Holliday, Estudió las múltiples e irreversibles causas del envejecimiento, utiliza datos de estudios comparativos entre diferentes especies de mamíferos y aves para analizar la relación entre el envejecimiento y la pérdida de mantenimiento, se hace referencia a la realización de análisis comparativos detallados de parámetros específicos, Se observa que las especies longevas tienden a tener mecanismos de reparación más eficientes y defensas antioxidantes más robustas en comparación con las especies de vida corta [11]. Baztán, et. al., el objetivo de este trabajo es evaluar las características de los pacientes asociadas a la ganancia funcional y estancia en las unidades geriátricas demedia estancia, se estudió a todos los pacientes ingresados

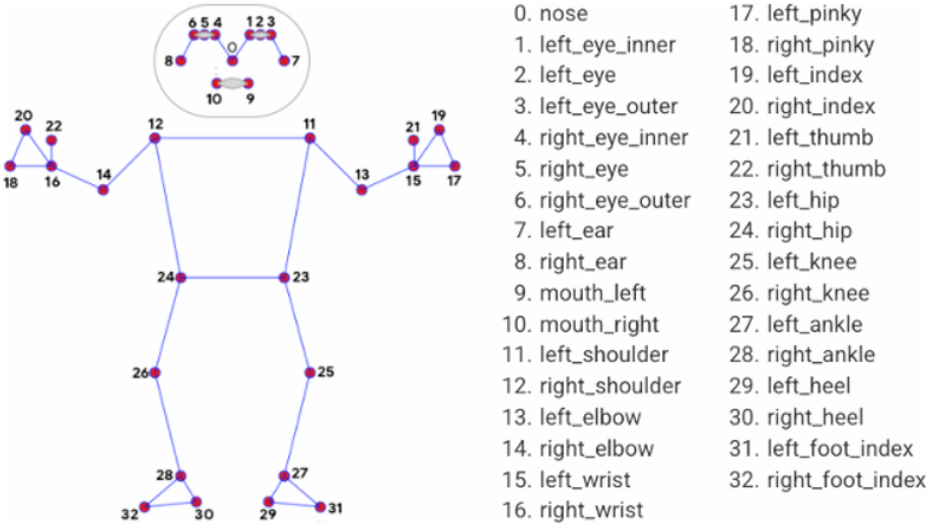


Fig. 2. Topología del cuerpo humano basado en google mediaPipe.

entre mayo de 2000 y diciembre de 2001. Se evaluó la ganancia funcional semanal y global con el Índice de Barthel. La estancia hospitalaria en unidades geriátricas de media estancia es adecuada, al menos, en las tres primeras semanas [2]. García, et. al, Demostrar las consecuencias unidas a las características socio- demográficas como afectan la calidad de vida de los mayores, con los datos de la II Encuesta Nacional de Factores de Riesgo de Enfermedades no Transmisibles de Cuba en el 2001. El efecto confusor de las variables se controló con un modelo de regresión logística, el modelo de regresión logística multivariada genera un 95 % de confiabilidad en un aumento de años al tener una situación económica favorables y pertenecer al sexo femenino [9].

Carmona, et. al, Licenciados en Rehabilitación en Salud para la atención geriátrica sean capaces de mejorar su desempeño profesional y como ser humano, Cuba, entre los países de América Latina y el Caribe se considera como uno de los más envejecidos (20,8 %) al cierre del 2019 con dos millones 307 mil 647 personas de 60 años y más, se propone la profesionalización conlleva el mejoramiento profesional y humano del licenciado en Rehabilitación en Salud a partir de la preparación permanente y continuada, permitirá a los adultos mayores se reincorporen a la sociedad de forma precoz o a las actividades comunes de la vida diaria [8].

Zaleski, et. al, Mostraron los beneficios de la actividad física regular o el ejercicio con respecto al envejecimiento, en los Estados Unidos, con más de 47 millones de adultos $\geq$ 65 años. Se proyecta que para el año 2030, el número de personas de 65 años o más alcanzará los 74 millones, La campaña alienta a los proveedores de atención médica a registrar la actividad física como un signo vital y prescribir ejercicio como lo harían, se demostró la eficacia del ejercicio para mejorar la salud y tratar enfermedades crónicas [19]. Moreno, et. al, la finalidad es comparar los resultados de dos Unidades con distinto enfoque en la rehabilitación de la fractura de cadera, estudio prospectivo de 286 pacientes mayores de 65 años, intervenidos de fractura de cadera durante los años 1997-2001.

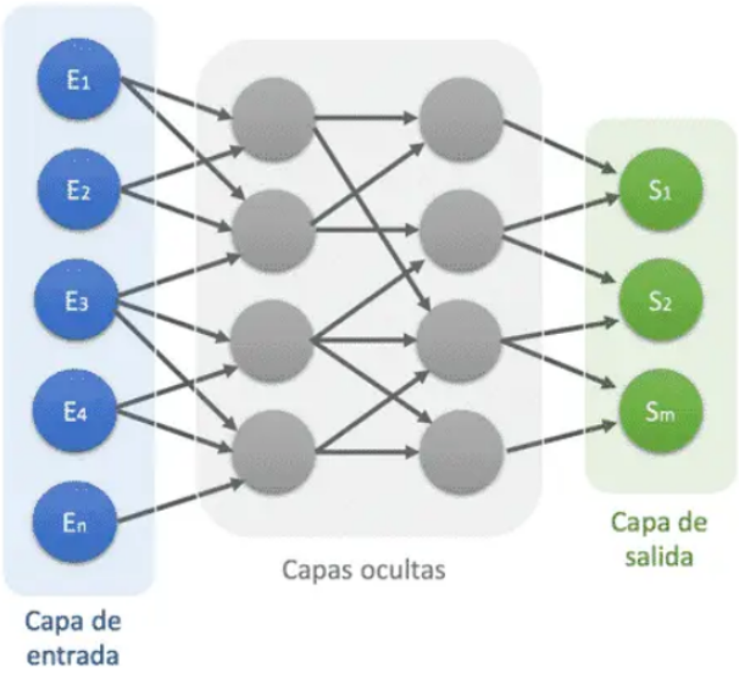


Fig. 3. Red neuronal artificial [3].

Se analizaron qué factores se asociaban con recuperación de la independencia mediante un análisis de regresión logística múltiple, La atención de las fracturas de cadera en Unidades de Rehabilitación Geriátrica mejora los resultados funcionales durante los primeros meses, pero se igualan a partir del 6.º mes [15]. Romero and Mora, Mostraron los beneficios de la rehabilitación geriátrica multidisciplinar en el paciente con fractura de cadera y demencia.

En el estudio canadiense Seitz et. al, de 11.200 pacientes con demencia y fractura de cadera, rehabilitación multidisciplinar es posible y que los resultados son mejores que la rehabilitación convencional, el modelo muestra ser más eficaz que el convencional por la recuperación funcional y menor tasa de institucionalización menores complicaciones de reingreso y mortalidad. [18]. La investigación toma como base la incorporación de las técnicas y tecnologías, por medio de la Inteligencia Artificial que integrará en los sistemas de rehabilitación geriátrica para el avance en la salud de las personas mayores con problemas de movilidad en el cuerpo, implementando algoritmos de aprendizaje automático y análisis de datos.

2.2. Profilaxis

La profilaxis en geriatría es la aplicación de medidas preventivas que disminuyen la vulnerabilidad acumulada por riesgos genéticos, estilos de vida y enfermedades. Dichas medidas deben definirse y llevarse a cabo desde una edad temprana. Se encamina a la prevención de enfermedades y a la protección de la salud del individuo.

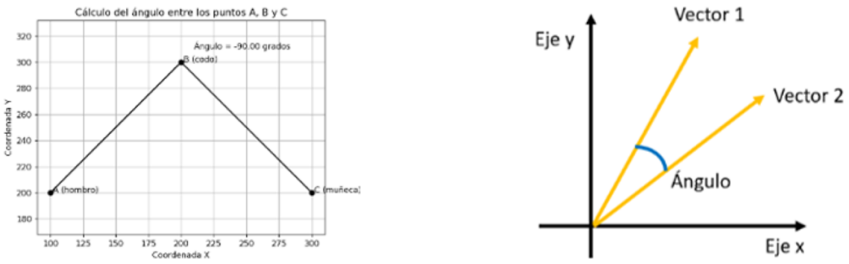


Fig. 4. Cálculo de los ángulos en base a las coordenadas y vectores.

La OMS define la profilaxis como "el proceso de optimización de oportunidades de la salud, la participación y seguridad con el fin de mejorar la calidad de vida a medida que las personas envejecen". La profilaxis en geriatría también puede incluir la prevención de enfermedad tromboembólica venosa (ETV), que son más frecuentes en los adultos mayores y pueden requerir profilaxis farmacológica eficaz [17].

2.3. Puntos de referencia del cuerpo humano

Los puntos de referencia del cuerpo humano consisten en la localización de las articulaciones a partir de una imagen o una secuencia de imágenes de una persona. Esencialmente, es una forma de capturar los ángulos por medio de puntos fundamentales como: muñecas, hombros, rodillas, ojos, tobillos y brazos. Permitiendo describir la pose de una persona.

2.4. Google mediapipe

MediaPipe Pose es una solución de aprendizaje automático para el seguimiento de posturas corporales de alta fidelidad, infiere 33 puntos de referencia y una máscara de segmentación de fondo en todo el cuerpo a partir de cuadros de video RGB. Los enfoques actuales de la última generación se basan principalmente en potentes entornos de escritorio para la inferencia, mientras que nuestro método logra un rendimiento en tiempo real en la mayoría de los teléfonos móviles, computadoras de escritorio y portátiles modernos, en Python e incluso en la web [14]. BlazePose es una arquitectura de red neuronal convolucional ligera para la estimación de la pose humana, está diseñada para la inferencia en tiempo real en dispositivos móviles. Durante la inferencia, la red produce 33 puntos clave del cuerpo para una sola persona. El hecho de que sea una red ligera capaz de funcionar en dispositivos móviles la hace particularmente adecuada para casos de uso en tiempo real, como el seguimiento del estado físico y el reconocimiento del lenguaje de señas [14].

2.5. Red neuronal convolucional

Son sistemas de programación robustos que facilitan principalmente la identificación de imágenes, asignando de manera automática a cada imagen suministrada en la entrada y una etiqueta correspondiente a la categoría a la que pertenece. Su método de operación es simple: el usuario aporta una imagen de entrada en formato de matriz de píxeles. Esta tiene 3 dimensiones:

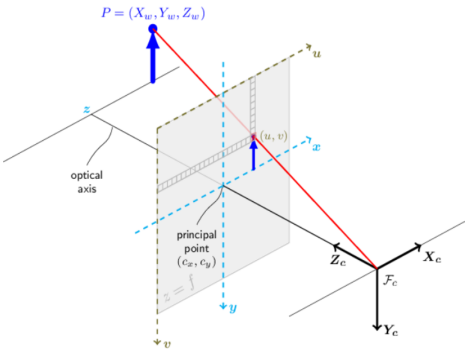


Fig. 5. Modelo de cámara estenopeica [16].

– Dos dimensiones para una imagen en niveles de gris.

– Una tercera dimensión, de profundidad 3 para representar los colores fundamentales (rojo, verde, azul) [6].

2.6. Funciones trigonométricas

Dentro de un algoritmo CNN utilizamos funciones trigonométricas para la obtención de coordenadas y medidas de los ángulos entre los puntos de referencia. Las interpretaciones de las fórmulas matemáticas que se utilizaron en el cálculo de los ángulos son las siguientes diferencias en coordenadas:

– Las diferencias en coordenadas se calculan como:

$$\Delta y_{BA} = y_A - y_B, \quad \Delta x_{BA} = x_A - x_B, \tag{1}$$

$$\Delta y_{BC} = y_C - y_B, \quad \Delta x_{BC} = x_C - x_B. \tag{2}$$

Las tangentes de los ángulos entre los puntos son:

$$\text{Tangente}_{BA} = \frac{\Delta y_{BA}}{\Delta x_{BA}}, \quad \text{Tangente}_{BC} = \frac{\Delta y_{BC}}{\Delta x_{BC}}. \tag{3}$$

La diferencia de ángulos utilizando la función de arco tangente de dos parámetros es:

$$\text{Diferencia} = \arctan\left(\frac{\text{Tangente}_{BC} - \text{Tangente}_{BA}}{1 + \text{Tangente}_{BA} \times \text{Tangente}_{BC}}\right). \tag{4}$$

Para ajustar el ángulo en el rango de 0 a 360 grados, se aplica la siguiente condición:

$$\text{si } \theta < 0, \text{ entonces } \theta = \theta + 360. \tag{5}$$

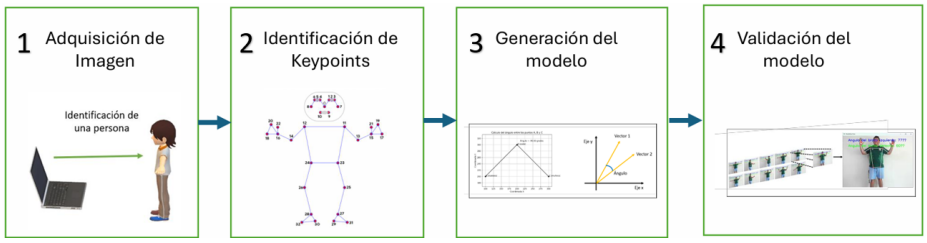


Fig. 6. Metodología del proceso.

2.7. Calibración de cámara

Las funciones de esta sección utilizan el llamado modelo de cámara estenopeica y el resultado es el punto 3D de una escena Pw en el plano de la imagen, mostrando una perspectiva del píxel correspondiente p. Ambos Pw y p se representan en coordenadas homogéneas, es decir, como vector homogéneo 3D y 2D respectivamente. A continuación se muestra la transformación proyectiva sin distorsión dada por un modelo de cámara estenopeica:

$$sp = A[R|t]P_w, \tag{6}$$

donde Pw es un punto 3D expresado con respecto al sistema de coordenadas mundial, p es un píxel 2D en el plano de la imagen, A es la matriz intrínseca de la cámara, R y t son la rotación y traslación que describen el cambio de coordenadas del mundo a la cámara (o marco de cámara) y s es la escala arbitraria de la transformación proyectiva y no forma parte del modelo de la cámara [16].

3. Materiales y métodos

Dado a los fundamentos teóricos presentados anteriormente, el diseño del modelo que debe incluirse en el identificador inteligente incluyen los siguientes elementos:

1. Adquisición de imagen: como se muestra en la figura 7. Se debe considerar la distancia correcta para que el identificador tome la lectura correcta.
2. Identificación de puntos clave de la postura: Como se muestra en la figura 2. Al tomar la imagen posteriormente se realiza la comparación de la imagen con el identificador de keypoints.
3. Generación del modelo: El modelo aplica las diferentes funciones trigonométricas para la obtención de los ángulos.
4. Validación del Modelo: Al obtener los cálculos de los ángulos se visualizara en la pantalla, como se muestra en la figura 4.

Los componentes que se utilizaron para el desarrollo del modelo fueron seleccionados tras investigación previa, siendo elementos importantes dentro de la Inteligencia Artificial.

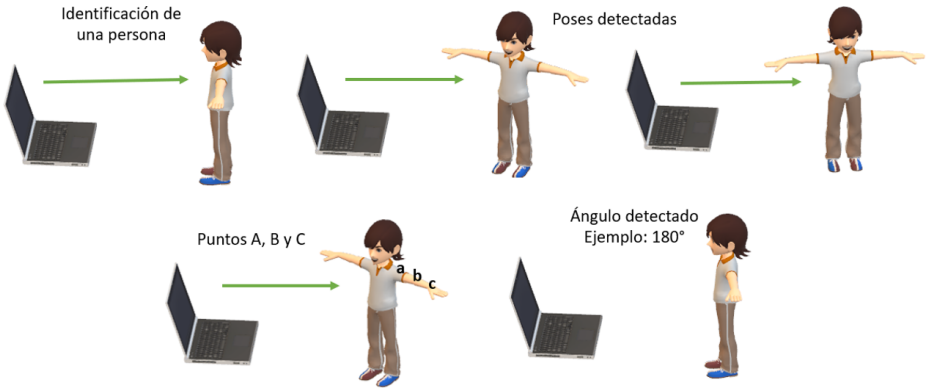


Fig. 7. Secuencia de funcionamiento del identificador de ángulos.

El uso de estas técnicas y tecnologías dentro de una rehabilitación geriátrica en adultos mayores de más de 60 años con problemas de movilidad simple o inclusive con indicios de dolor en brazos podría demostrar el mejoramiento progresivo del paciente además de llevar un control de ejercicios para cada usuario dedicando mínimo 30 minutos al día para lograr mejores resultados. Para poder empezar a identificar una imagen y encontrar una persona, se emplea un detector de posición basado en la cara. Siendo que se parte de la premisa que es la parte del cuerpo humano más fácil y menos costosa de utilizar. Por lo que cuando comenzamos a utilizar la técnica blazepose además de identificar la cara o el rostro humano, nos daría como punto central la cadera, la circunferencia de la persona y el ángulo que tiene el tronco del cuerpo. Todo el código utilizado dentro del desarrollo del identificador de ángulos se implementó en el lenguaje de programación Python.

3.1. Blaze pose

Dado que BlazePose reproduce los 33 puntos clave del cuerpo humano, resulta ser un buen elemento para casos de uso en tiempo real, como lo es la identificación y seguimiento de los movimientos físicos y de las posturas del cuerpo. Este algoritmo de machine learning es nuestra principal técnica para poder realizar los cálculos de los ángulos dentro de puntos de referencia principales. Consta de dos modelos de aprendizaje automático: un detector y un estimador. El detector recorta la región humana de la imagen de entrada, mientras que el estimador toma una imagen de resolución de 256×256 de la persona detectada como entrada y genera los puntos clave [5].

3.2. Implementación

Indagando diferentes maneras de utilizar la percepción visual a favor de la investigación y en el desarrollo del identificador, se emplearon las librerías de OpenCV comúnmente utilizadas en el área de visión por computadora, aprendizaje automático y procesamiento de imágenes además de la librería MediaPipe donde utiliza algoritmos de

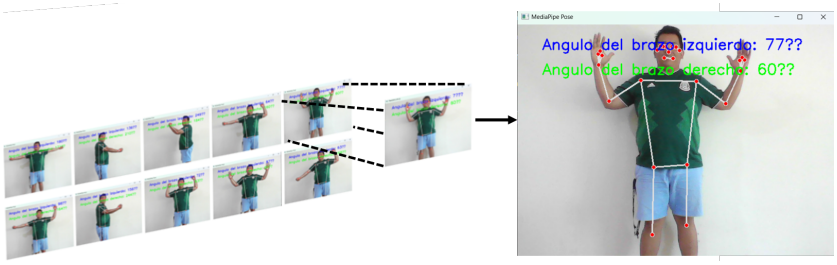


Fig. 8. Representación de movimientos mediante poses utilizando el identificador de ángulos.

aprendizaje automáticos para detectar y dar seguimiento a la postura del usuario a través de una cámara web en tiempo real. Para la identificación de puntos clave de postura, solo se utilizaron tres puntos representando las articulaciones de los hombros, codos y muñecas, los cuales al calcular el ángulo entre ellos, se determina la posición del brazo y la orientación en relación con el cuerpo del usuario. En este proceso la identificación se realiza con la librería mediapipe, utilizando redes neuronales convolucionales(CNN).

Luego de poder detectar la imagen dentro de la cámara web, se realiza el cálculo de los ángulos utilizando funciones trigonométricas y la función arco tangente de dos parámetros. En este caso, utilizaremos solo como referencia esos puntos clave para detectar el ángulo de movimientos en los brazos utilizando coordenadas para las poses detectadas, podemos realizar el cálculo de ángulos definiendo una función en código Python donde cuyas tangentes (x, y, z) calculan el ángulo en radianes para una mejor interpretación. Tomando como punto central el punto B (codo), el punto A (hombro) y el punto C (muñeca) tal como se muestra en la figura 7 Obteniendo de la línea formada por el punto medio y el punto final, como se muestra en la figura 4 sustituyendo los dos radianes para convertirlos en el ángulo que formarían los tres puntos de referencia.

Al utilizar la librería de mediapipe, se inicializa una entrada de vídeo usando cv2 que usualmente comienza desde el punto 0 solo si se utiliza un dispositivo de entrada, y se asignarán los parámetros necesarios en la función que realizará los cálculos. Luego de leer el fotograma se procede a la parte de la ubicación de los puntos de referencia, mostrará la información de los cálculos en la pantalla y finalmente se detendrá el ciclo de ejecución del programa. Por lo que importaremos las librerías necesarias para este proceso desde Python:

– import cv2
– import mediapipe as mp
– import math

Se comenzó con el funcionamiento de la cámara y detección de poses, usando una función trigonométrica utilizada para la técnica de blaze pose en el cálculo de los ángulos con los tres puntos de referencia ya definidos, descrito anteriormente en la figura 8. Para poder aplicar la identificación de ángulos solo tendremos que ejecutar el programa desarrollado y que el usuario pueda hacer los movimientos solicitados. Donde, se utilizó una Raspberry Pi 4 como medio de pruebas del identificador de ángulos, cargando el código en Python y ejecutando las pruebas piloto

en usuarios geriátricos con movilidad..... esto devolverá la información de los tres puntos detectados ya mencionados siguiendo la codificación programada anteriormente. Cualquier persona que quiera reproducir el experimento en cuestión puede ponerse en contacto con los autores de la investigación.

3.3. Validaciones

Para poder asegurar que el identificador cumpla con el objetivo principal de la investigación y del funcionamiento del mismo, además de cumplir con las necesidades básicas del usuario final se debe validar un antes y un después de su aplicación. Proponiendo los siguientes puntos o pasos para llevar una correcta evaluación:

1. Validación interna: Para este punto, la evaluación se realizó por medio de un profesional en el área de fisioterapia donde efectuamos las siguientes tareas: actualización de información de movimientos básicos en un paciente geriátrico (profilaxis), aplicación en la marcha en los usuarios para recuperación de movilidad en el cuerpo humano, detección de ejercicios en las demás extremidades, vinculación de ejercicios con personas con fracturas y facilidad de uso del usuario.
2. Validación externa: Se realizó a través de usuarios finales, quienes llevaron a cabo los movimientos básicos de rehabilitación en los brazos para medir el rendimiento de la aplicación. Esto se llevó a cabo con un grupo de usuarios participantes en pruebas piloto, quienes proporcionaron retroalimentación sobre la efectividad y la usabilidad de la aplicación en situaciones reales de rehabilitación.

4. Resultados

Dadas las coordenadas resultantes por las posturas de los usuarios, podemos calcular los ángulos de diversas partes del cuerpo mediante los puntos de referencia a, b, c. Ya que los cálculos se realizan dentro de un espacio donde la orientación no afecta en las poses del usuario y de las deducciones finales, se simularon las mediciones de ángulos en cinco usuarios diferentes generando coordenadas de acuerdo a los movimientos realizados por cada uno. Sabemos que la forma del cuerpo humano y las longitudes de las mismas varían dependiendo las personas, no se podría asignar una distancia entre puntos de referencia por qué no serían los mismos resultados de las posturas realizadas. Pero al efectuar el cálculo de distancia entre la muñeca, hombro y codo es muy probable que la distancia sea similar entre varias posiciones. Comenzamos evaluando nuestros resultados y el método propuesto conforme los usuarios finales utilizaban el identificador, obteniendo puntos importantes a considerar para las actualizaciones futuras como lo son:

– Variaciones en puntos de vista y poses del usuario
– Investigaciones previas sobre los puntos de referencia, y el desarrollo de un modelo de aprendizaje.

5. Conclusión

En este artículo, se ha propuesto el desarrollo de un identificador de ángulos en puntos de referencia del cuerpo humano utilizando machine learning, específicamente con el uso de herramientas tecnológicas como google mediaPipe. El objetivo ha sido proporcionar una solución precisa y de bajo costo para la estimación de partes y posturas corporales, especialmente enfocada en personas geriátricas. Además de la estimación de ángulos, hemos identificado un vasto potencial en la extracción de información adicional utilizando esta tecnología. Como resultado, proponemos varias áreas de continuación y futuros trabajos:

– Ampliar la investigación a otras categorías de objetos para aumentar la versatilidad y utilidad de la herramienta, además de indagar sobre los cálculos de puntos de referencia de otras articulaciones del cuerpo humano, por ejemplo rodillas y rotación de caderas donde la movilidad de los adultos mayores afectan con mayor frecuencia.
– Aumentar la base de datos mediante pruebas adicionales en usuarios finales para mejorar la precisión y generalización del modelo.
– Explorar técnicas de aprendizaje profundo para obtener resultados aún más precisos en la detección de imágenes y reconocimiento de patrones.
– Realizar pruebas piloto en usuarios con diferentes problemas de movilidad o condiciones físicas, para poder obtener distintos tipos de resultados y variables que nos puedan servir como apoyo en el campo de la investigación.

El desarrollo de un identificador de ángulos utilizando el algoritmo BlazePose de machine learning tiene una respuesta favorable durante las pruebas piloto y validaciones. Además de que la funcionalidad del mismo va representada de dispositivos de bajo costo con acceso a cualquier usuario que le interese adquirir el sistema concluido e implementado en este tipo de pequeños ordenadores.

Las actualizaciones continuas, que incluyen ajustes en el diseño y mejoras en el contenido audiovisual, respaldan nuestra elección de técnicas y herramientas de desarrollo al alinear nuestras soluciones con las expectativas y necesidades del usuario. Estas puntos en la investigación son esenciales para avanzar de manera significativa en este campo y asegurar una continuación lógica y efectiva del proyecto [4].

Referencias

1. Baztán, J., González, M., Morales, C., Vázquez, E., Morón, N., Forcano, S., Ruipérez, I.: Variables asociadas a la recuperación funcional y la institucionalización al alta en ancianos ingresados en una unidad geriátrica de media estancia. Revista clinica española, vol. 204, no. 11, pp. 574–582 (2004) doi: 10.1016/S0014-2565(04)71550-7
2. Baztán, J. J., Domenech, J. R., González, M., Forcano, S., Morales, C., Ruipérez, I.: Ganancia funcional y estancia hospitalaria en la unidad geriátrica de media estancia del hospital central de cruz roja de Madrid. Revista española de salud pública, vol. 78, pp. 355–366 (2004)
3. Calvo, D.: Definición de red neuronal artificial (2017) https://www.diegocalvo.es/definicion-de-red-neuronal/

4. Chung, J. L., Ong, L. Y., Leow, M. C.: Comparative analysis of skeleton-based human pose estimation. Future Internet, vol. 14, no. 12, pp. 380 (2022) doi: 10.3390/fi14120380
5. Cochard, D.: Blazepose : A 3D pose estimation model (2021) https://medium.com/axinc-ai/blazepose-a-3d-pose-estimation-model-d8689d06b7c4
6. Datascientest: Convolutional neural network: Definición y funcionamiento (2024) https://datascientest.com/es/convolutional-neural-network-es
7. Ferrer, B. C., Olivares, D. Y., Chisholm, D. H.: Algunas consideraciones bioéticas en la rehabilitación geriátrica (2020)
8. Ferrer, B. C., Olivares, D. Y. R., Hernández, D.: Importancia de la profesionalización en rehabilitación geriátrica (2020)
9. García-Roche, R. G., Sánchez, M. H., Pérez, P. V., de la Rosa, M. C., Gorbea, M. B., Álvarez, S. S.: Calidad de vida relacionada con la salud de los adultos mayores en el país, 2001. Revista Cubana de Higiene y Epidemiología, vol. 48, no. 1, pp. 43–52 (2010)
10. Gobierno de México: Gerontología, geriatría y adultos mayores (2018) https://www.gob.mx/salud/articulos/gerontologia-geriatria-y-adultos-mayores
11. Holliday, R.: Aging: The reality: The multiple and irreversible causes of aging. The Journals of Gerontology Series A: Biological Sciences and Medical Sciences, vol. 59, no. 6, pp. B568–B572 (2004) doi: 10.1093/gerona/59.6.B568
12. Instituto Nacional de las Personas Adultas Mayores: Beneficios de la actividad física en los adultos mayores (2018) https://www.gob.mx/inapam/articulos/beneficios-de-la-actividad-fisica-en-los-adultos-mayores?idiom=es
13. Mayelin, L. S., Dayami, G. V.: La música en la rehabilitación geriátrica. In: Primera Jornada Nacional Virtual (2021)
14. Mediapipe: Pose (2024) https://google.github.io/mediapipe/solutions/pose
15. Moreno, J., García, I., Serra, J., Núñez, C., Bellón, J., Álvarez, A.: Estudio comparativo de dos modelos de rehabilitación en las fracturas de cadera. Rehabilitación, vol. 40, no. 3, pp. 123–131 (2006) doi: 10.1016/S0048-7120(06)74878-2
16. OpenCV: Camera calibration and 3D reconstruction (2024) https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html
17. Organización Mundial de la Salud: profilaxis en geriatria (2017) https://www.who.int/es/news-room/questions-and-answers/item/profilaxis-en-geriatria
18. Romero-Pisonero, E., Mora Fernández, J.: Rehabilitación geriátrica multidisciplinar en el paciente con fractura de cadera y demencia. Revista Española de Geriatría y Gerontología, vol. 54, no. 4, pp. 220–229 (2019)
19. Zaleski, A. L., Taylor, B. A., Panza, G. A., Wu, Y., Pescatello, L. S., Thompson, P. D., Fernandez, A. B.: Coming of age: Considerations in the prescription of exercise for older adults. Methodist DeBakey cardiovascular journal, vol. 12, no. 2, pp. 98 (2016) doi: 10.14797/mdcj-12-2-98

ISSN 1870-4069

Filtros Espaciales aprendibles en CNNs: Análisis de filtros de Gabor hacia un entrenamiento más eficiente

Carlos Orozco-Solis[1], Alfonso Rojas-Domínguez[1], Héctor Puga[1], Manuel Ornelas Rodríguez[1] Martín Carpio[1], Valentín Calzada-Ledesma[2]

[1] Tecnológico Nacional de México, Campus León, México

[2] Tecnológico Nacional de México, campus Purísima del Rincón, México

alfonso.rojas@gmail.com

Resumen. La integración de filtros de Gabor en redes convolucionales (CNNs) ha sido objeto de estudio debido a su capacidad para capturar características relevantes en imágenes. En este trabajo, se investiga el impacto de estos filtros en el rendimiento y la estabilidad del entrenamiento de una CNN. Se realizan dos experimentos utilizando bloques GCC (Gabor Convolutional) y CCC (Convolutional), variando la entrenabilidad y la inicialización de los bloques. Se analiza la convergencia de los parámetros de los filtros de Gabor a lo largo del entrenamiento y se emplea un enfoque de clusterización para identificar patrones de comportamiento. Los resultados muestran que la inclusión de filtros de Gabor no degrada significativamente el desempeño de la red y que estos filtros se estabilizan rápidamente, lo que permitiría reducir el tiempo de entrenamiento de las redes. Se presenta un análisis detallado de los comportamientos de los filtros y se discute cómo diferentes configuraciones de los bloques mencionados afectan la precisión de la red. Estos hallazgos proporcionan información valiosa para la optimización de redes neuronales convolucionales que integren filtros espaciales entrenables.

Palabras clave: Filtros entrenables, filtros de Gabor, redes convolucionales.

Learnable Spatial Filters in CNNs: Analysis of Gabor Filters Towards a More Efficient Training

Abstract. The integration of Gabor filters in convolutional neural networks (CNNs) has been the subject of study due to their ability to capture relevant features in images. This work investigates the impact of these filters on the performance and stability of CNN training. Two experiments are conducted using GCC (Gabor Convolutional) and CCC (Convolutional) blocks, varying the trainability and initialization of the blocks. The convergence of Gabor filter

parameters during training is analyzed, and a clustering approach is employed to identify behavior patterns. The results show that the inclusion of Gabor filters does not significantly degrade the network's performance and that these filters stabilize quickly, potentially reducing network training time. A detailed analysis of filter behaviors is presented, and the discussion covers how different block configurations affect network accuracy. These findings provide valuable insights for the optimization of convolutional neural networks that integrate trainable spatial filters.

Keywords: Trainable spatial filters, Gabor filters, CNNs.

1. Introducción

Las Redes Neuronales Convolucionales (CNNs por sus siglas en inglés) [8] poseen una poderosa capacidad de aprendizaje, que, a través de la implementación de múltiples etapas de extracción de características, les permite aprender automáticamente representaciones complejas de datos [4], particularmente en imágenes digitales. Antes de la popularización de las CNNs, el proceso de extracción de características en imágenes se realizaba mediante filtros espaciales, los cuales eran diseñados a mano por un experto en el campo.

Éstos ofrecen una gran flexibilidad y adaptabilidad, ya que pueden detectar una amplia gama de características, tales como bordes, esquinas, texturas, entre otros. Sin embargo, existe un problema fundamental, y es que para extraer las características principales de una o un conjunto de imágenes, es necesario tener un conocimiento a priori del dominio del problema para elegir cuidadosamente los filtros necesarios para resaltar las características deseadas, este proceso en sí mismo puede ser muy laborioso y exhaustivo, ya que implica ajustar manualmente los parámetros de cada filtro.

Las CNNs automatizan el proceso de extracción de características a través de la aplicación de capas convolucionales, cuyos parámetros se establecen automáticamente mediante un proceso de entrenamiento. En la literatura se ha observado que tras entrenar una CNN (como AlexNet [6]), los filtros convolucionales de la primera capa tienden hacia la creación de filtros espaciales, similares a los filtros de Gabor [10], los cuales son efectivos para extraer atributos simples, tales como bordes, esquinas y texturas. Asimismo, en [12] se afirma que los pesos de capas iniciales de diversos modelos de imágenes tienden a converger hacia filtros de Gabor, entre otros, y que las características universales observadas con más frecuencia en modelos de imágenes son funciones de base canónica 2D de Fourier, como filtros de Gabor o wavelets.

Es decir, los atributos que son aprendidos en las primeras capas convolucionales pueden ser obtenidos mediante filtros espaciales. Así, en la literatura se propuso reemplazar algunos filtros convolucionales con filtros espaciales, conduciendo al desarrollo de varias estrategias revisadas a profundidad en la Sección 2. Destaca un tipo de arquitectura llamada GaborNet la cual implementa filtros de Gabor en la primera capa convolucional, que llamaremos la capa de Gabor. Aunque la GaborNet ha sido utilizada en la literatura, la falta de un análisis exhaustivo de sus parámetros limita su comprensión y la capacidad para maximizar su rendimiento.

Tabla 1. Hiperparámetros de los experimentos.

Parámetro	Valor	Descripción
Epochs	100	Número de iteraciones del entrenamiento.
Batch size	1024	Tamaño del lote de entrenamiento.
Learning rate	0.001	Taza de aprendizaje para el entrenamiento.
Executions	35	Número de repeticiones del experimento.
Optimizer	Adam	Algoritmo de optimización utilizado.
Loss	Cross-entropy	Función de pérdida para optimizar el modelo.

Hasta donde sabemos, en la literatura no hay un análisis que permita comprender del todo dicha arquitectura. Es por ello que el presente trabajo se enfoca en llenar este vacío, mediante un análisis detallado de los parámetros de la capa de Gabor. Este análisis revela hallazgos significativos para mejorar la eficacia y eficiencia de las GaborNet. Por ejemplo, se observa que el diseño del filtro se decide de manera temprana durante las primeras iteraciones del proceso de entrenamiento, lo que sugiere la importancia de una inicialización adecuada. Éste y otros hallazgos avanzan hacia una base sólida para futuras investigaciones en el diseño y optimización de CNNs que incorporan filtros espaciales, y destacan la importancia de un análisis detallado de los parámetros para maximizar el rendimiento de estos modelos en aplicaciones prácticas.

2. Trabajos relacionados

Para aprovechar los beneficios de los filtros espaciales, algunos autores han propuesto aplicarlos sobre imágenes de entrenamiento antes de pasarlas a una CNN. En [7], aplican filtros espaciales para detectar bordes y esquinas en imágenes faciales, mejorando la precisión obtenida por una CNN en 8.5 %. En [3], proponen usan filtros espaciales para la clasificación de dígitos escritos a mano, logrando resultados competitivos contra LeNet [8], que obtiene 99.05 % de clasificación correcta, mientras que la propuesta alcanza 99.16 %. La integración de filtros espaciales en CNNs es una tendencia reciente que ha demostrado mejorar el rendimiento en algunos casos.

Por ejemplo, la red GCN [10] utiliza un banco de filtros de Gabor con U orientaciones y V escalas, logrando 99.37 % de precisión en MNIST, comparable al 99.43 % de la Red de Respuesta Orientada [16]. Ambas redes utilizan 0.25 y 0.49 millones de parámetros, respectivamente. En CIFAR10/100, la GCN alcanza 96.12 % y 79.87 %, respectivamente, reduciendo a la mitad los parámetros requeridos en comparación con la Wide Residual Network (WRN) [15], que logra 96.00 % y 80.95 %.

Para integrar aún más los filtros espaciales en las CNNs, se ha propuesto que la primera capa convolucional esté compuesta exclusivamente de filtros espaciales, ajustados por la red de la misma manera que las capas convolucionales convencionales. Por ejemplo, GaborNet [1] y PCFNet [11] emplean este enfoque. En ambos trabajos, se observa que la ventaja de rendimiento entre el uso de filtros espaciales en la primera capa o no usarlos disminuye a medida que aumenta el tamaño del conjunto de datos de entrenamiento.

Tabla 2. Configuración de los experimentos y resultados.

	Bloque GCC	**Bloque CCC**	**Accuracy**
Exp.	**(Gabor)**	**(Convolución)**	**en la Prueba**
1	Entrenable	Entrenable	77.55 %
2.a	Entrenable	Ausente	61.17 %
2.b	Pre-entrenado	Entrenable	67.32 %

Esto se demuestra en pruebas realizadas en una CNN de Filtros Predefinidos (PCF) [11], que implementa filtros espaciales como Gabor (Ga), Sobel (So) y Schmid (Sc) en su capa inicial. Utilizando solo el 5 % del conjunto de datos CIFAR10, PCF-GaSc-ResNet18 logró una precisión del 66.32 %, superando a ResNet18 (con un 62.05 %). De manera similar, utilizando solo el 20 % del conjunto de datos CIFAR100, PCF-GaSc-WRN168 alcanzó un 55.15 %, superando a WRN168 (con un 53.27 %). Sin embargo, al evaluar los conjuntos de datos CIFAR10/100 completos, no se observaron diferencias significativas. En [1], se comparan una CNN simple y AlexNet con sus respectivas variantes de Gabor. La Gabor-CNN supera a la CNN en 6 % de precisión en el problema "Dogs vs Cats". En el conjunto de datos "AffectNet", la diferencia es de 3 %, y en "ImageNet", no se observa una diferencia significativa.

3. Metodología

3.1. Redes neuronales convolucionales

Una Red Neuronal Convolucional (CNN) es un tipo de red neuronal profunda diseñada para procesar datos organizados en una cuadrícula, típicamente imágenes. Las CNNs aprenden representaciones jerárquicas de datos a través de sus capas convolucionales [9]. Las representaciones aprendidas luego se emplean en tareas de análisis de imágenes como clasificación, segmentación, reconocimiento de identidad, etc. Una CNN está organizada en varias capas, cada una de las cuales contiene múltiples filtros o kernels convolucionales.

Estos kernels se aplican a las imágenes de entrada mediante convolución, una operación que implica el desplazamiento de una ventana sobre la imagen (1). Este proceso facilita la extracción de características al utilizar un conjunto específico de pesos que se multiplican por los elementos correspondientes del campo receptivo [2]. La operación de convolución puede expresarse de la siguiente manera:

$$g(x,y) = w * f(x,y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i,j) f(x-i, y-j), \tag{1}$$

donde $g(x,y)$ es la imagen filtrada, $f(x,y)$ es la imagen original, w es el kernel convolucional, $\sum_{i=-a}^{a} \sum_{j=-b}^{b}$ denota una suma doble sobre todas las posiciones del kernel, los índices i y j representan coordenadas en el kernel, y a y b definen el tamaño del núcleo (normalmente $a = b$).

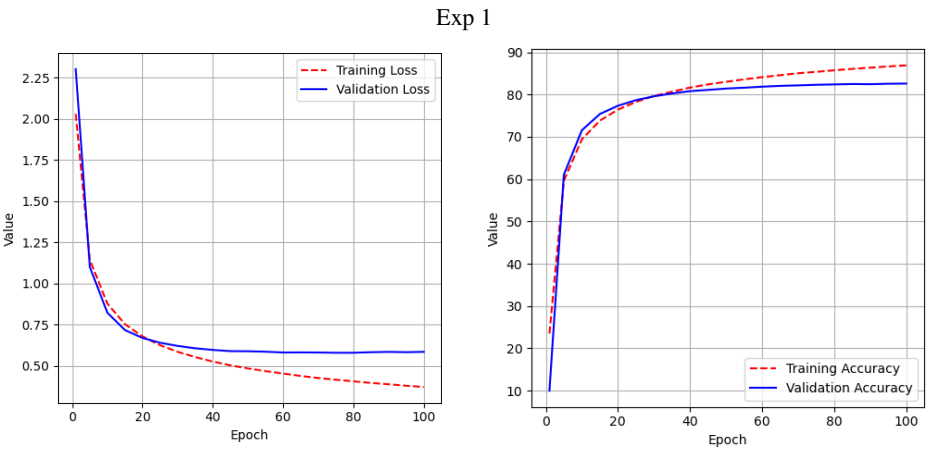


Fig. 1. Curvas de pérdida (Izq.) y precisión (Der.) durante el entrenamiento y validación del modelo - experimento 1.

3.2. Arquitectura

Este estudio emplea dos variantes de la misma arquitectura. La primera variante consiste en dos bloques, GCC y CCC: el bloque GCC incluye una capa Gabor con 40 filtros de tamaño 7 y dos capas convolucionales con 64 y 128 filtros de tamaño 3, respectivamente. El bloque CCC consta de tres capas convolucionales con 256 filtros de tamaño 3. En la segunda variante, el bloque GCC permanece igual, pero el bloque CCC se reemplaza por una operación de MaxPooling, lo que permite que las dimensiones de los mapas de activación se ajusten a la primera capa Fully Connected (FC). En total, la red contiene tres capas FC.

Si se utilizara una primera capa convolucional tradicional con 40 filtros w, cada uno de tamaño 7×7, ello implicaría un total de 49 parámetros que la red debería entrenar por cada filtro es decir, $40 \times 49 = 1960$ parámetros en total. Dada la tendencia de estos filtros a converger hacia patrones espaciales similares a funciones de Gabor, reemplazamos los filtros w con funciones de Gabor, que son sinusoides complejas moduladas por una envolvente Gaussiana [1]:

$$g(x, y, w, \theta, \psi, \sigma) = \exp\left(-\frac{x'^2 + y'^2}{2\sigma^2}\right) \exp\left(i(wx' + \sigma)\right), \tag{2}$$

$$x' = \quad x\cos(\theta) + y\sin(\theta), \tag{3}$$

$$y' = -x\cos(\theta) + y\sin(\theta). \tag{4}$$

La ecuación (2) se puede expresar en sus partes real e imaginaria; en este trabajo, utilizamos la parte real de la función Gabor:

$$g(x, y, w, \theta, \psi, \sigma) = \exp\left(-\frac{x'^2 + y'^2}{2\sigma^2}\right) \cos(wx' + \psi), \tag{5}$$

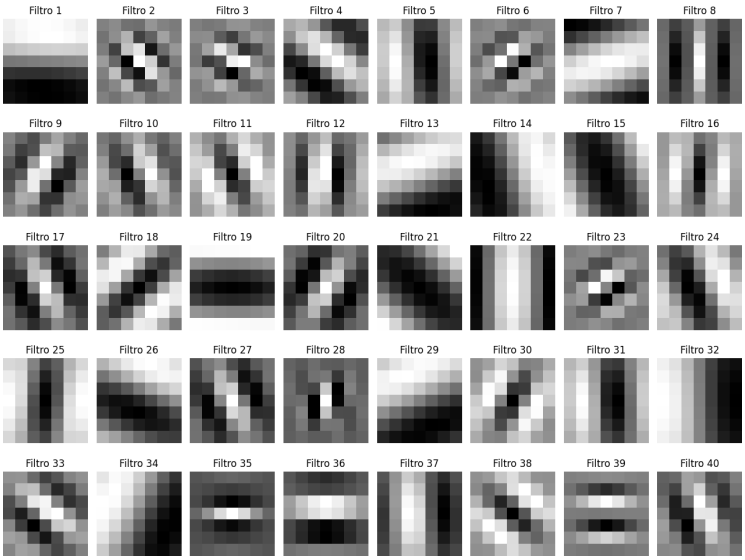


Fig. 2. Filtros de la capa Gabor entrenados - experimento 1.

donde (x, y) denota la posición del píxel en el dominio espacial, w representa la frecuencia angular central de una onda plana sinusoidal, θ indica la rotación en sentido contrario a las agujas del reloj de la función Gaussiana (es decir, la orientación del filtro de Gabor), σ representa la desviación estándar de la Gaussiana, y ψ controla la fase de las oscilaciones sinusoidales dentro del filtro de Gabor.

Establecemos $\sigma \approx \pi/w$ para definir la relación entre σ y w como se describe en [13]. Al combinar múltiples filtros de Gabor con diferentes orientaciones y frecuencias, se forma un banco de filtros. La capa Gabor es una capa convolucional especializada, diseñada para aplicar un banco de filtros de Gabor a través del cual los 49 parámetros para un filtro ahora se reducen a solo 4 parámetros: $(w, \theta, \psi, \sigma)$ ($40 \times 4 = 160$ parámetros en total para la red). Integrar esta capa de Gabor en la CNN produce la arquitectura GaborNet [1]. Debido a que modificamos la GaborNet original, y para evitar confusiones, nos referimos a nuestra propia implementación como **GaborNet2**.

3.3. Implementación

Para desarrollar **GaborNet2** en Python, utilizamos la librería PyTorch, que proporciona una interfaz dinámica para construir y entrenar modelos de aprendizaje profundo. Con esta herramienta, podemos implementar de manera efectiva una arquitectura de red neuronal convolucional (CNN) en la que se integra una capa con filtros de Gabor. La arquitectura base de la CNN en PyTorch se define creando una clase que hereda de `nn.Module`. En el constructor de esta clase se definen las capas convolucionales (`nn.Conv2d`), de agrupación (`nn.MaxPool2d`), de normalización (`nn.BatchNorm2d`), y de activación (`nn.ReLU`, `nn.Sigmoid`, etc.). Una característica notable de PyTorch es la integración de varios optimizadores, como el descenso de gradiente estocástico (SGD), Adam, RMSprop o Adagrad.

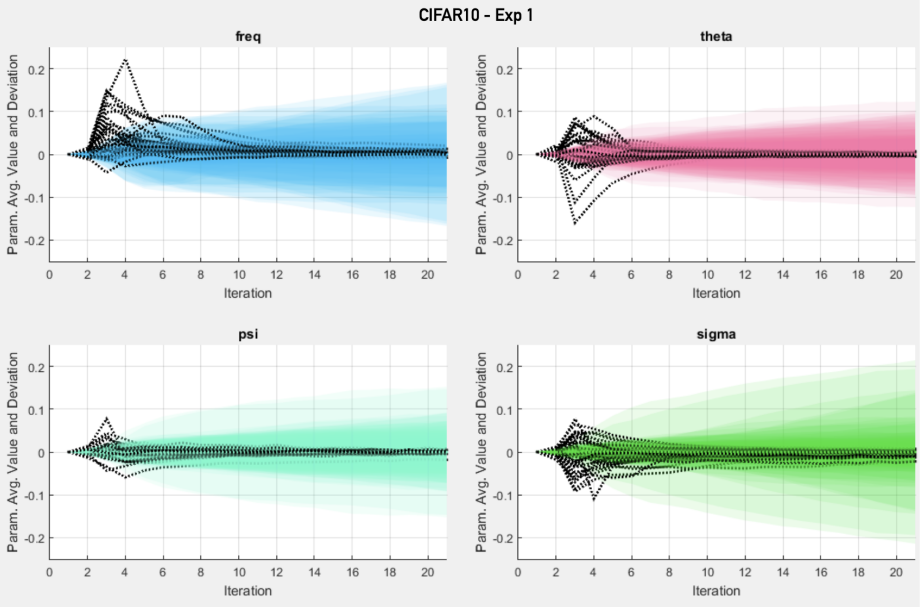


Fig. 3. Evolución de parámetros de la capa Gabor durante el entrenamiento - experimento 1. Se muestran simultáneamente las curvas y regiones sombreadas de las 35 repeticiones sombreadas.

Estos optimizadores ajustan los pesos de la red para minimizar una función de pérdida mediante la actualización iterativa de los pesos siguiendo el descenso de gradientes calculados automáticamente. Por esta razón, la capa de Gabor se diseñó basándose en la capa tradicional `nn.Conv2d`.

La capa convolucional `nn.Conv2d` es una subclase de `_ConvNd`, por lo que la capa personalizada `GaborConv2d` también hereda de ésta. En el constructor de esta capa personalizada, se definen 4 parámetros principales de los filtros de Gabor: frecuencia espacial, orientación, desviación estándar de la Gaussiana y fase. Estos parámetros son entrenables y se actualizan mediante SGD (con el optimizador Adam) de la misma manera que los pesos de una capa tradicional. Una vez definida la capa `GaborConv2d`, es fácil reemplazar la primera capa convolucional de la arquitectura base de la CNN con esta capa personalizada. Al hacerlo, la CNN utilizará filtros de Gabor en lugar de filtros convencionales en su capa inicial.

4. Configuración experimental

En esta sección, se detallan los parámetros y la configuración utilizados para evaluar la convergencia de los filtros de **GaborNet2**. Para comprender mejor el comportamiento de la red, se llevaron a cabo dos experimentos principales utilizando variantes de la arquitectura. Los hiperparámetros usados se muestran en el Cuadro 1 y la configuración de experimentos en el Cuadro 2. El Experimento 1 corresponde a la utilización de la primera variante de la arquitectura, entrenando la red de manera convencional.

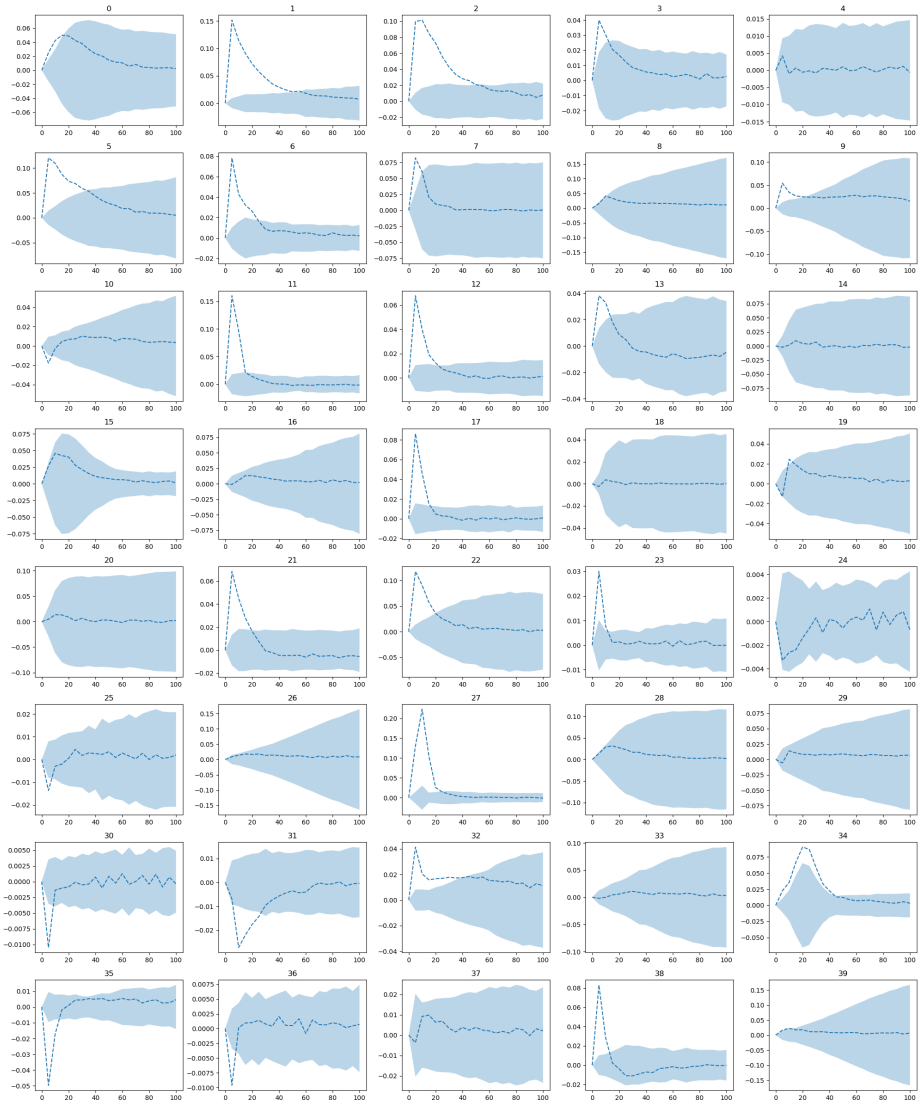


Fig. 4. Evolución de los 40 valores de frecuencia w - experimento 1.

En el Experimento 2, se separó el entrenamiento en dos etapas para obtener un análisis más detallado. En la etapa a), se empleó la segunda variante de la arquitectura para dar mayor énfasis a la capa de Gabor y observar de manera más precisa el comportamiento de los parámetros (w, θ, ψ, σ) durante el entrenamiento. Una vez finalizado este proceso, se guardaron los pesos obtenidos; posteriormente, en la etapa b) se retomó la primera variante de la arquitectura, cargando los pesos resultantes de la etapa a) y congelando el primer bloque de la red (para no perder los pesos ya entrenados hasta ese punto).

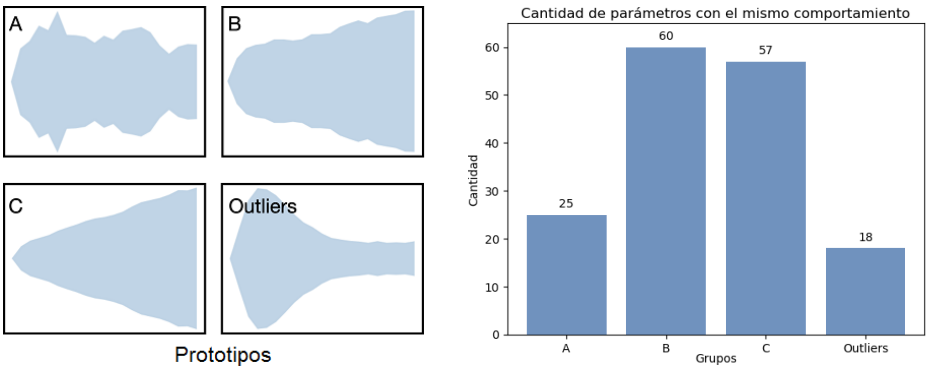


Fig. 5. Representantes (Izq.) y su frecuencia (Der.) - experimento 1.

Luego se continuó el entrenamiento de la red hasta finalizar con las épocas de entrenamiento. Con el Experimento 2 buscamos promover la mejor convergencia de los filtros de la capa Gabor, al entrenarlos de manera aislada de las otras capas convolucionales, y además buscamos determinar si el rendimiento obtenido es comparable al del Experimento 1 (donde se entrenaron ambos bloques de la red simultáneamente). Es importante destacar que cada experimento se repitió 35 veces utilizando la misma inicialización, garantizando así que cada filtro tuviera los mismos valores en la época 0, para las 35 repeticiones.

Parámetros del Filtro de Gabor: La inicialización de los 4 parámetros $(w, \theta, \psi, \sigma)$ de la capa de Gabor se basa en [13]:

$$w_n = \frac{\pi}{2} \cdot 2^{\frac{-(n-1)}{2}}, \tag{6}$$

$n \in \{1, ..., 5\}$; $\theta_m = \frac{\pi}{8} \cdot m - 1$, $m \in \{1, ..., 8\}$; $\psi \sim \mathcal{U}(0, \pi)$; y $\sigma \approx \frac{\pi}{w}$. Esta configuración resulta en un banco de filtros de Gabor con 5 escalas y 8 orientaciones diferentes. Esta diversidad en escalas y orientaciones permite una gran adaptabilidad para una amplia gama de características en las imágenes.

Dataset: CIFAR-10 es un conjunto de datos ampliamente utilizado en el campo del aprendizaje automático para entrenar y probar modelos de clasificación de imágenes. El conjunto de datos consta de 60,000 imágenes en color de 32×32 píxeles, distribuidas uniformemente en 10 clases (6,000 por clase) [5]. Para este trabajo, las imágenes se convirtieron a escala de grises y se separaron aleatoriamente en tres conjuntos: 40,000 imágenes para entrenamiento, 10,000 para validación y 10,000 para prueba.

5. Resultados

En la Fig. 1 se pueden observar las curvas de pérdida de entrenamiento y pérdida de validación para el Experimento 1. La curva de pérdida de entrenamiento (linea punteada roja) muestra una disminución constante a lo largo de las 100 épocas, mientras que la curva de precisión aumenta de manera continua.

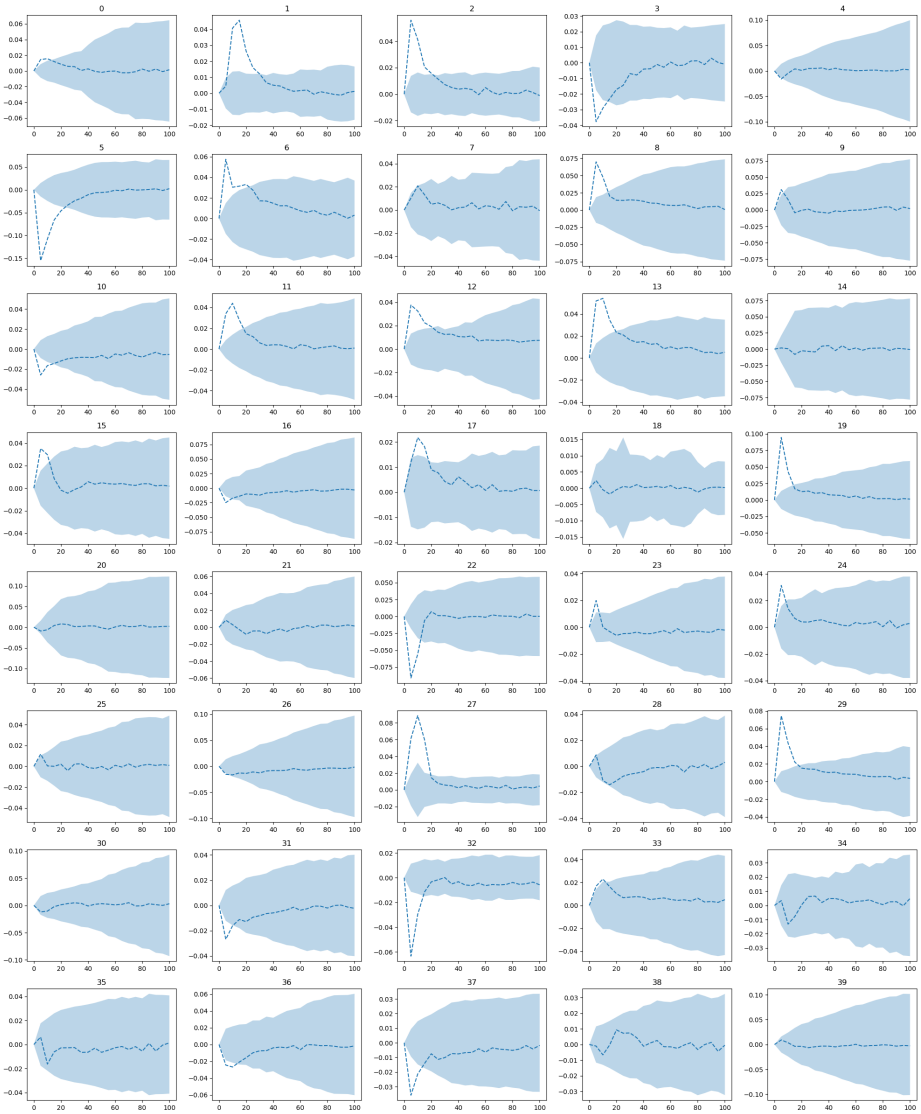


Fig. 6. Evolución de los 40 valores de orientación θ - experimento 1.

Esto sugiere un buen proceso de entrenamiento y que los filtros de la capa Gabor se integran adecuadamente con el resto de la arquitectura de la red. Los filtros de la capa Gabor entrenados se muestran en la Fig. 2. Para evaluar la convergencia de los filtros, se guardaron los valores de los parámetros $(w, \theta, \psi, \sigma)$ de la capa Gabor cada 5 épocas, durante el entrenamiento, así como los valores iniciales. El resultado se presenta en la Fig. 3, donde las líneas punteadas muestran la diferencia de la media (sobre las 35 repeticiones realizadas) de cada parámetro en esa época con respecto a su media en la época anterior.

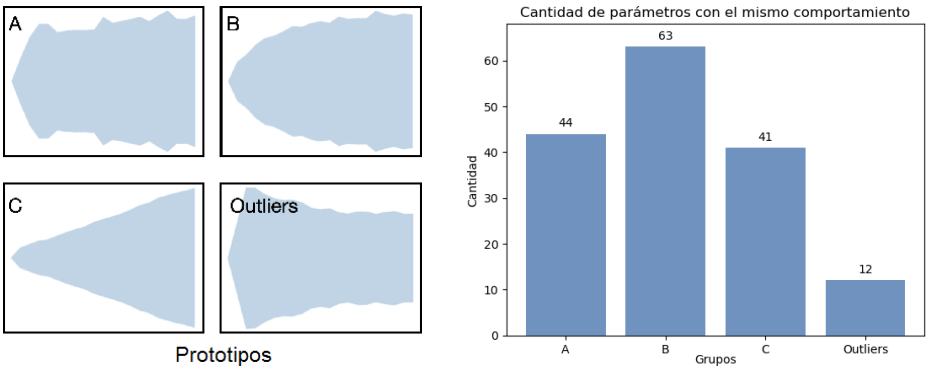


Fig. 7. Representantes (Izq.) y su frecuencia (Der.) - experimento 2.

Cuando esa línea sube, indica que en esa época el parámetro aumentó su valor con respecto a la época anterior; por el contrario, si la línea baja, indica que el valor del parámetro correspondiente disminuyó. Las regiones sombreadas representan las desviaciones estándar de cada parámetro. Esta representación visual nos permite analizar la estabilidad y convergencia de los filtros a lo largo del entrenamiento de manera precisa y detallada. En la Fig. 3, se puede apreciar que en conjunto los parámetros de la capa Gabor se estabilizan alrededor de la época 50 para el Experimento 1, esto sugiere que el entrenamiento de la capa Gabor puede detenerse antes de las 100 épocas, ya que antes de eso se llega a un punto en el que los parámetros ya no cambian significativamente.

Por otro lado, para observar la convergencia de los parámetros, es crucial centrarse en las desviaciones estándar, pues éstas reflejan la variación de los valores de los parámetros por cada época. Dado que en la Fig. 3 los filtros están superpuestos, dificultando su análisis, se muestran los filtros por separado en la Fig. 4 (para frequencia) y la Fig. 6 (para θ). Por restricciones de espacio evitamos mostrar las gráficas de los otros dos parámetros.

Analizar individualmente 160 gráficas de comportamiento (40 filtros y 4 parámetros) es muy laborioso e ineficiente, por ello, se decidió agrupar los filtros con comportamientos similares mediante un proceso de clusterización. Este enfoque permitió identificar patrones comunes entre los filtros y simplificar su análisis e interpretación de resultados. Para el proceso de clusterización se consideró solamente la información de desviaciones estándar, se aislaron los comportamientos atípicos (outliers) que pudieran sesgar los resultados y se identificaron interdependencias entre las variables del conjunto de datos.

Dadas la presencia de fuertes interdependencias, se consideró útil normalizar los datos y aplicar un Análisis de Componentes Principales (PCA) para reducir la información redundante y extraer las características más relevantes. Se concluyó que cuatro componentes son suficientes para explicar la mayor parte de la varianza. En seguida, para determinar el número óptimo de clusters, se utilizó la técnica llamada Gap Statistic, que considera la diferencia entre la suma de las dispersiones intraclase y la suma de las dispersiones interclase [14]. Se evaluó un rango de dos a seis clusters, e inicialmente se identificó el número óptimo de clusters igual a seis.

Posteriormente se aplicó el algoritmo K-means para clusterizar los datos en función de los 4 componentes principales seleccionados. Sin embargo, al analizar visualmente los resultados, se observó que algunos clusters aún mostraban gran similitud entre ellos, y tras unir los clusters similares se obtuvo como resultado final tres grupos distintos, además de los conjuntos de outliers. El proceso descrito arriba condujo a la obtención de grupos homogéneos y representativos de los comportamientos de los parámetros durante el entrenamiento. El mismo proceso se realizó de manera independiente sobre los datos del Experimento 1 y del Experimento 2.

La Fig. 5, ilustra los comportamientos representativos de cada grupo (A, B y C), así como la frecuencia de ocurrencia de los filtros, además del conjunto de outliers, para el Experimento 1. En seguida se describen brevemente cada grupo de comportamientos obtenido. El grupo A se caracteriza por tener oscilaciones pronunciadas, indicando que no hay una convergencia del parámetro a lo largo del entrenamiento. Los grupos B y C son relativamente similares, pues ambos muestran que la dispersión de los valores de los parámetros aumenta conforme avanza el entrenamiento (esto es contrario a la convergencia esperada).

Sin embargo, en el grupo B, a diferencia del C, se observan ligeras oscilaciones. En cuanto a los casos atípicos, se destaca el mostrado en la Fig. 5, donde se observa la convergencia del parámetro (en las primeras épocas la desviación incrementa hasta alcanzar un máximo y luego disminuye gradualmente); éste es el comportamiento deseado, pues muestra que el parámetro alcanzó un valor muy similar en las 35 repeticiones del experimento, reflejando estabilidad y convergencia efectiva en este caso específico.

También en la Fig. 5 se muestra la frecuencia de ocurrencia de los filtros con comportamiento dentro de cada grupo. Los outliers tienen la menor cantidad (18), de los cuales solo 2 corresponden al caso mencionado anteriormente. Los grupos B y C tienen más del doble de filtros que la clase A, es decir, en 73 % de los filtros la desviación estándar aumenta con las épocas. Este hallazgo sugiere que en cada una de las 35 repeticiones puede encontrarse un valor óptimo diferente para los parámetros de un filtro, resultando beneficioso para la red en general.

Para el Experimento 2, como ilustra la Fig. 7, los comportamientos de los parámetros pueden resumirse de forma muy similar a la del Experimento 1, indicando consistencia en los grupos identificados previamente como representativos del comportamiento de los parámetros durante el entrenamiento de la red. Destacando nuevamente los casos atípicos, en la Fig. 7 se presenta otro ejemplo en el que el parámetro muestra convergencia a lo largo del entrenamiento, aunque en este caso, el descenso de la desviación estándar no es tan pronunciado como en el ejemplo dado para el Experimento 1.

Al igual que antes, se muestra la ocurrencia de comportamientos por cada grupo, y se destaca que la mayor parte de los filtros muestran oscilaciones. De entre ellos, 44 no tienen una clara tendencia (clase A) y 63 muestran un comportamiento creciente además de oscilaciones (clase B). Finalmente, en el Cuadro 2, se presenta un resumen de los resultados obtenidos en los dos experimentos realizados. En el Experimento 1, donde se emplearon bloques GCC y CCC entrenables, se alcanzó un accuracy de 77.55 % en los datos de prueba.

Por otro lado, en el Experimento 2.a, se mantuvo el bloque GCC entrenable pero se eliminó el bloque CCC, lo que resultó en un accuracy de 61.17 %. En el Experimento 2.b, se utilizó un bloque GCC pre-entrenado y un bloque CCC entrenable, logrando un accuracy del 67.32 %. Estos resultados muestran claramente cómo diferentes configuraciones de bloques afectan el rendimiento de la red en términos de precisión en la clasificación de datos de prueba.

6. Conclusión

Los resultados obtenidos en este estudio destacan que la inclusión de filtros de Gabor en la arquitectura de una red neuronal, no conllevan a una degradación significativa del rendimiento de la red. Por el contrario, se observa que estos filtros se estabilizan rápidamente durante el proceso de entrenamiento, alcanzando una convergencia efectiva en aproximadamente 50 épocas. Este hallazgo podría ser prometedor en términos de eficiencia y eficacia para el desarrollo de modelos de aprendizaje profundo. Sin embargo, es necesario realizar más estudios para comprobar si este comportamiento es replicable en otros escenarios y contextos.

El análisis detallado de la convergencia de los filtros de Gabor se llevó a cabo mediante un enfoque de clusterización, permitiendo agrupar los filtros con comportamientos similares durante el proceso entrenamiento, simplificando así la interpretación de los resultados y proporcionando una visión más clara de la estabilidad de los filtros, esto a través de los diferentes contextos y condiciones de entrenamiento establecidos en nuestro diseño de experimentos.

Los resultados de la clusterización revelaron la existencia de tres grupos distintos de comportamientos de los filtros de Gabor, así como casos atípicos, que representan situaciones particulares en la convergencia de los parámetros. Aproximadamente el 69 % (73 % Experimento 1 y 65 % Experimento 2) de los filtros exhiben un aumento en la desviación estándar a lo largo del entrenamiento, indicando variaciones significativas en los valores de los parámetros. Esta observación es crucial, ya que sugiere que en cada repetición del experimento es posible encontrar un valor óptimo diferente para los parámetros de un filtro, lo cual podría ser beneficioso para la red, al poder explorar una gama más amplia de configuraciones para su entrenamiento.

Estos hallazgos brindan una comprensión más profunda del cómo los filtros de Gabor interactúan con una CNN y cómo su estabilidad influye en el rendimiento global. La repetibilidad de comportamientos y patrones de convergencia en diferentes experimentos sugiere que los efectos observados en los filtros de Gabor son consistentes y reproducibles, lo cual es fundamental para la validez y futura aplicabilidad de estos resultados, así como aplicaciones prácticas en aprendizaje automático y visión artificial.

7. Trabajo futuro

Se planea realizar futuras investigaciones para explorar la generalización de los resultados obtenidos en este estudio mediante la evaluación en diferentes conjuntos de datos. Al aumentar la diversidad de los datos de entrada, se respaldarán las conclusiones de este trabajo en distintos escenarios y dominios.

Además, se tiene previsto investigar el rendimiento de la capa Gabor en diferentes arquitecturas. Comparar varias arquitecturas permitirá determinar la escalabilidad y la adaptabilidad de los filtros Gabor en las CNNs. También se contempla la posibilidad de probar otros filtros espaciales además del filtro Gabor, lo cual ampliará la comprensión sobre qué filtros mejoran el rendimiento de las CNNs. Estas investigaciones planificadas contribuirán a enriquecer y fortalecer aún más los hallazgos presentados en este estudio.

Agradecimientos. Este trabajo fue apoyado por el Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) de México, a través de la Beca de Posgrado 824517 (C. Orozco) y el proyecto CÁTEDRAS-2598 (A. Rojas).

Referencias

1. Alekseev, A., Bobe, A.: Gabornet: Gabor filters with learnable parameters in deep convolutional neural network. In: International Conference on Engineering and Telecommunication, pp. 1–4 (2019) doi: 10.1109/EnT47717.2019.9030571
2. Bouvrie, J.: Notes on convolutional neural networks (2006)
3. Calderon, A., Roa, S., Victorino, J.: Handwritten digit recognition using convolutional neural netorks and Gabor filters. Proceedings of the International Congress on Computational Intelligence, pp. 1–9 (2003)
4. Khan, A., Sohail, A., Zahoora, U., Qureshi, A. S.: A survey of the recent architectures of deep convolutional neural networks. Artificial intelligence review, vol. 53, pp. 5455–5516 (2020) doi: 10.1007/s10462-020-09825-6
5. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Toronto (2009)
6. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, vol. 25 (2012) doi: 10.1145/3065386
7. Kwolek, B.: Face detection using convolutional neural networks and Gabor filters. In: International Conference on Artificial Neural Networks, pp. 551–556 (2005) doi: 10.1007/11550822_86
8. LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., Jackel, L. D.: Backpropagation applied to handwritten zip code recognition. Neural computation, vol. 1, no. 4, pp. 541–551 (1989) doi: 10.1162/neco.1989.1.4.541
9. Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J.: A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems, vol. 33, no. 12, pp. 6999–7019 (2021) doi: 10.1109/TNNLS.2021.3084827
10. Luan, S., Chen, C., Zhang, B., Han, J., Liu, J.: Gabor convolutional networks. IEEE Transactions on Image Processing, vol. 27, no. 9, pp. 4357–4366 (2018) doi: 10.1109/TIP.2018.2835143
11. Ma, Y., Luo, Y., Yang, Z.: PCFnet: Deep neural network with predefined convolutional filters. Neurocomputing, vol. 382, pp. 32–39 (2020) doi: 10.1016/j.neucom.2019.11.075
12. Marchetti, G. L., Hillar, C., Kragic, D., Sanborn, S.: Harmonics of learning: Universal fourier features emerge in invariant networks. In: The Thirty Seventh Annual Conference on Learning Theory, pp. 3775–3797 (2023)
13. Meshgini, S., Aghagolzadeh, A., Seyedarabi, H.: Face recognition using gabor filter bank, kernel principle component analysis and support vector machine. International Journal of Computer Theory and Engineering, vol. 4, no. 5, pp. 767 (2012)

14. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology), vol. 63, no. 2, pp. 411–423 (2001) doi: 10.1111/1467-9868.00293
15. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv (2016)
16. Zhou, Y., Ye, Q., Qiu, Q., Jiao, J.: Oriented response networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 519–528 (2017)

ISSN 1870-4069

Síntesis dimensional óptima de un mecanismo para seguimiento de trayectoria por medio de búsqueda armónica y evolución diferencial

Alvaro Sánchez-Márquez, Silvia Sánchez-Márquez,
Josefina Hernández-Tapia, Alberto Hernández-Lazcano,
Leonel Sergio Carrasco-Pérez, Claudia Alicia Romero-León

Universidad Autónoma de Tlaxcala
Unidad Académica Multidisciplinaria,
Campus Calpulalpan,
México

{asanchez, silvia.sanchez, jhernandezt, ahlazcano
20012178, 20154466}@uatx.mx

Resumen. En este trabajo se presenta la síntesis dimensional de un mecanismo de cuatro barras para el seguimiento de una trayectoria definida geométricamente como semicírculo, dicho caso de estudio se plantea como un problema de optimización numérico con restricciones. Es importante mencionar que este tipo de problemas se vuelven más complejos cuando las restricciones aumentan, sin embargo es resuelto por algoritmos heurísticos, específicamente búsqueda armónica y evolución diferencial. Aun considerando la complejidad dada por el número de restricciones de este caso de estudio (25 en nuestro caso); los resultados obtenidos determinan las proporciones de los eslabones para el mecanismo, se consideran satisfactorios en base al error obtenido que es de precisión 10^{-6}.

Palabras clave: Síntesis dimensional, optimización, búsqueda armónica, evolución diferencial.

Optimal Dimensional Synthesis of a Mechanism for Trajectory Tracking Through Harmonic Search and Differential Evolution

Abstract. This work presents the dimensional synthesis of a four-bar mechanism for tracking a trajectory geometrically defined as a semicircle. This case study is formulated as a numerical optimization problem with restrictions. It is important to mention that kind of problems become more complex as the restrictions increase; however, it is solved by heuristic algorithms, specifically harmonic search and differential evolution. Even considering the complexity given by the number of restrictions of this case study (25 in our case); the results obtained determine the proportions of the links for the mechanism, they are considered satisfactory based on the error obtained, which is precision 10^{-6}.

Keywords: Dimensional synthesis, optimization, harmonic search, differential evolution.

1. Introducción

El hombre ha mostrado su ingenio para idear máquinas y mecanismos con el objetivo de tener una vida más fácil y cómoda, las máquinas son dispositivos que se utilizan al modificar, transmitir y dirigir fuerzas para llevar a cabo un objetivo específico [12]. El diseño de éstos mecanismos se enfoca en principalmente tres técnicas: síntesis de tipo (clase de mecanismo seleccionado), síntesis cuantitativa o analítica (número de elementos del mecanismo) y la síntesis dimensional (longitud y ángulo de los elementos del mecanismo), el mayor interés en este trabajo radica en ocuparse en la técnica de síntesis dimensional debido a que se busca las dimensiones óptimas de un mecanismo de cuatro barras para el seguimiento de una trayectoria específica.

Los problemas de síntesis dimensional se pueden resolver mediante dos métodos: los métodos de programación matemática (también llamados clásicos) y los métodos heurísticos (específicamente los algoritmos metaheurísticos), los cuales son procedimientos de búsqueda y han demostrado ser capaces de obtener una buena solución (aunque no garantizan encontrar el valor óptimo) a problemas difíciles.

Estos métodos imitan fenómenos simples observados en la naturaleza, destacan algoritmos genéticos (Genetic Algorithm, GA) [2], optimización basada en colonia de hormigas (Ant Colony optimization) [17], colonia artificial de abejas (Artificial Bee Colony, ABC) [20], algoritmo de luciérnagas (Firefly Algorithm, FA) [21], Evolución diferencial (Differential Evolution, ED) [16], Búsqueda Armónica (Harmony Search, HS) [22], por mencionar algunos.

El presente trabajo aborda el seguimiento de trayectoria por medio de búsqueda armónica, un algoritmo metaheurístico que basa su funcionamiento en el proceso de la improvisación de músicos de jazz, fue desarrollado por Geem et al. en 2001 [10]. Los algoritmos de búsqueda armónica han sido aplicados para una gran amplia variedad de trabajos tales como: identificación de parámetros para modelos de celdas solares [1], diseño óptimo de costes de las redes de distribución de agua [11], diseño geométrico óptimo de Cúpulas Geodésicas [15], composiciones musicales [9], entre otros.

Por otro lado, el algoritmo de evolución diferencial propuesto por Storn y Price [16] es un modelo evolutivo basado en la población, utilizado para resolver numerosos problemas complejos, algunas investigaciones con aplicación de este algoritmo metaheurístico están basadas en: la síntesis óptima de un mecanismo plano para seguimiento de trayectoria [18], la sintonización de clasificadores difusos para el reconocimiento del lenguaje de señas [19], control de un motor de corriente directa bajo incertidumbre paramétrica [8], diagnóstico de fallos en sistemas industriales [5], diseño automático de redes neuronales artificiales [7], por citar algunos.

Este proyecto de investigación aborda el análisis del mecanismo de cuatro barras, así como su cinemática; se determinan las particularidades del problema de optimización numérica y la trayectoria propuesta para el caso de estudio. Los algoritmos aplicados para resolver el problema de optimización son búsqueda armónica y evolución diferencial, cuyos resultados obtenidos muestran la mejora significativa con Evolución diferencial.

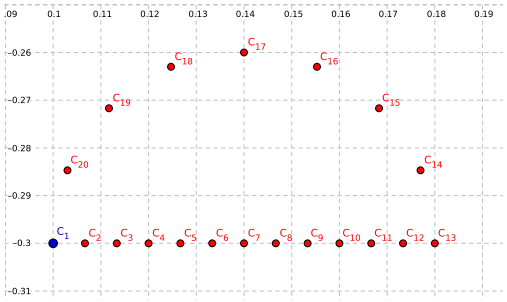


Fig. 1. Trayectoria deseada $(\overline{x}_E, \overline{y}_E)$.

2. Análisis del mecanismo

Un mecanismo de cuatro barras está formado por tres eslabones móviles y una barra fija, gráficamente se observa en la Figura 1. La longitud de cada una de las barras se representa por $l_i, i = 1, 2, 3, 4$, l_5 y l_6 son distancias estratégicas entre algunos vértices, el desplazamiento angular con respecto a la horizontal lo indica θ_i, r_{cx} y r_{cy} señalan la posición del acoplador C.

2.1. Cinemática del mecanismo

Para determinar la posición del punto C a lo largo de una trayectoria se realiza la síntesis del mecanismo a través del modelado geométrico, se calcula l_5 dado por (1) y por ley de cosenos el ángulo α (2):

$$l_5^2 = l_1^2 + l_2^2 - 2\ l_1\ l_2\ \cos(\theta_2), \tag{1}$$

$$\alpha = \cos^{-1}\left(\frac{l_3^2 + l_4^2 - l_5^2}{2\ l_3\ l_4}\right). \tag{2}$$

Los ángulos β y δ son calculados por ley de senos y se muestran en las ecuaciones (3) y (4):

$$\beta = \sin^{-1}\left(\frac{l_2\ \sin(\theta_2)}{l_5}\right), \tag{3}$$

$$\delta = \sin^{-1}\left(\frac{l_4\ \sin(\alpha)}{l_5}\right). \tag{4}$$

Sin embargo el ángulo δ está conformado por los ángulos θ_3 y β como se aprecia en la Figura **??**, por lo que la ecuación (1) consigue calcular el ángulo θ_3:

$$\theta_3 = \begin{cases} \delta + \beta\ \theta_2 \leq \pi, \\ \delta - \beta\ \theta_2 > \pi. \end{cases} \tag{5}$$

Tabla 1. Límites de variables de diseño.

	l_1	l_2	l_3	l_4	r_cx	r_cy	θ_0	x_0	y_0	θ_2^1	**...**	θ_2^20	$\overline{x}_ini$
P_max	0.5	0.5	0.5	0.5	0.5	0.5	2Π	60	60	2Π	...	2Π	0.5
P_min	0	0	0	0	0	0	0	-60	-60	0	...	0	-0.5

La longitud l_6 y el ángulo φ indicados en la Figura 1 corresponden a las ecuaciones (6) y (7):

$$l_6^2 = r_{cx}^2 + r_{cy}^2, \tag{6}$$

$$\varphi = tan^{-1}\left(\frac{r_{cy}}{r_{cx}}\right). \tag{7}$$

Una vez que se han encontrado los parámetros que se incluyen en el diagrama geométrico que se muestra en la la Figura 1, entonces, es posible determinar la posición geométrica del punto acoplador C y esta dada por:

$$\begin{bmatrix} C_x \\ C_y \end{bmatrix} = \begin{bmatrix} l_2\ \cos(\theta_0+\theta_2) + l_6\ \cos(\varphi+\theta_3+\theta_0) \\ l_2\ \sin(\theta_0+\theta_2) + l_6\ \sin(\varphi+\theta_3+\theta_0) \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}. \tag{8}$$

3. Trayectoria semicircular para el seguimiento

La trayectoria propuesta tiene una recta horizontal que parte desde el punto $C_1 = (\overline{x}_{ini}, \overline{y}_{ini})$ hasta el punto C_{13}, la mitad de la circunferencia esta dada por 7 puntos $C_{14}...C_{20}$, es decir, la trayectoria está formada por 20 coordenadas cartesianas, de las cuales 13 puntos forman la alineación horizontal y los 7 puntos restantes forman la mitad de la circunferencia (ver Figura 1). Dicha trayectoria fue propuesta en el trabajo [13] donde se utilizó un mecanismo planar de ocho eslabones con un grado de libertad como extremidad bípeda para el seguimiento similar a la marcha humana que está definida por el desplazamiento en forma de semicírculo. El vector solución p está dado por 29 variables de diseño presentado en (9):

$$p = [l_1, l_2, l_3, l_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, \theta_2^2, ...\theta_2^{20}]^T \in \mathbb{R}^{29}, \tag{9}$$

donde:

- l_1, l_2, l_3, l_4 corresponden a las longitudes de las barras,
- r_{cx}, r_{cy} indican la posición del acoplador,
- θ_0 es el ángulo de orientación del sistema con respecto a la horizontal,
- x_0, y_0 señalan el origen del sistema de referencia, y
- $\theta_2^1, ...\theta_2^{20}$ son los valores de los ángulos de la barra l_2 con respecto a los 20 puntos dados en la Figura 1.

Algoritmo 1.1: Algoritmo de búsqueda armónica.

definir función objetivo $f(x), x = (x_1, x_2, ..., x_N)$;
definir tasa de selección de la memoria armónica r_{accept};
definir tasa de ajuste de tono r_{pa};
definir rango de ajuste de tono bw;
generar memoria armónica inicial MA (arreglo $(k \times N)$);
while $g <$ máximo número de iteraciones **do**
    while $i <= N$ **do**
        if rand $< r_{\text{accept}}$ **then**
            indice $\leftarrow$ rand$(1, k)$;
            if rand $< r_{pa}$ **then**
                newH$(i) \leftarrow MA$(indice, $i) + bw *$ rand$(-1,\ 1)$;
            else
                newH$(i) \leftarrow MA$(indice, $i)$
        else
            newH$(i) \leftarrow$ rand$(L_i,\ U_i)$
    remplazar la nueva armonía por la peor de MA, si esta es mejor ;

3.1. Ley de Grashof

La Ley de Grashof afirma que la barra más corta de un mecanismo de cuatro barras da vueltas enteras respecto a todas las otras, si se cumple que la suma de la longitud de la barra más larga l y la de la más corta s es más pequeña o igual que la suma de las longitudes de las otras dos p y q [6], es decir:

$$s + l \leq p + q. \tag{10}$$

Cabe mencionar que el sistema mecánico planteado en este trabajo la Ley de Grashof esta dada por la ecuación 11:

$$l_1 + l_2 \leq l_3 + l_4. \tag{11}$$

3.2. Solución factible

La solución de un problema que satisface el total de las restricciones planteadas se conoce como **solución factible**. El planteamiento de restricciones obliga a considerar ciertos métodos o reglas para el manejo de las mismas, para este caso se utilizan las reglas de viabilidad de Deb [4].

1. Entre dos soluciones factibles, se selecciona la que tiene el mejor desempeño en la función objetivo.
2. Entre una solución factible y una no factible, se elige la solución factible.
3. Entre dos soluciones no factibles, se prefiere el que tiene la suma más baja de violaciones de restricciones.

Algoritmo 1.2: Algoritmo de evolución diferencial.

1 Generar aleatoriamente una población inicial de soluciones
  $x_{j,i} = b_{j,L} + (b_{j,U} - b_{j,L}) * \text{rand}(0,1)$;
2 Repetir;
3 Seleccionar un padre y tres individuos aleatoriamente $(X_{r_0,g},\ X_{r_1,g},\ X_{r_2,g})$;
4 Crear un hijo $U_{i,g}$ a partir de la cruza o recombinación entre vector padre $X_{i,g}$ y el vector de mutación $V_{i,g}$;
5 Evaluar la aptitud y factibilidad del hijo generado;
6 **if** $f(U_{i,g}) \leq f(X_{i,g})$ **then**
7 $\quad U_{i,g}$;
8 **else**
9 $\quad X_{i,g}$
10 Hasta que se satisfaga una condición de paro;

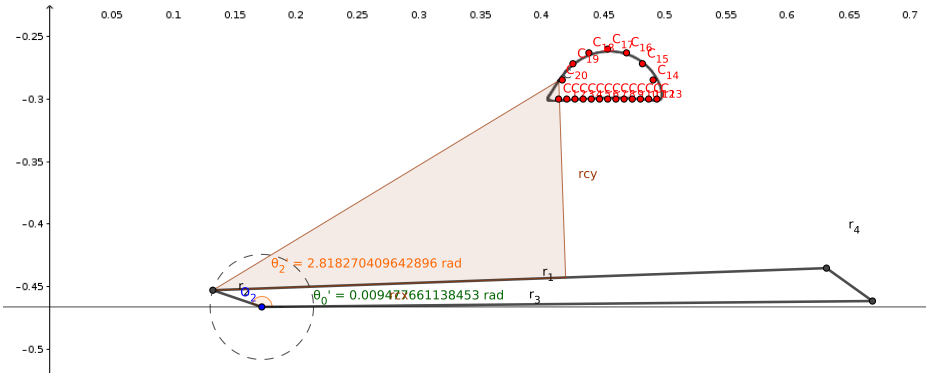


Fig. 2. Mecanismo óptimo: Búsqueda armónica.

4. Esquema de optimización

El problema de optimización numérico es descrito por las ecuaciones (12) a (13), el propósito es obtener la solución de la síntesis dimensional del mecanismo de cuatro barras que siga la trayectoria definida por el semicírculo que se muestra en la Figura 1.

4.1. Función objetivo

La función objetivo está dada por la suma de las diferencias cuadradas de la trayectoria deseada $(\overline{x}_E^i, \overline{y}_E^i)$ y la trayectoria obtenida (x_E^i, y_E^i), de las 20 coordenadas cartesianas (ecuación (12)):

$$\text{mín } f(p) = \sum_{i=1}^{\overline{n}=20} \left[(\overline{x}_E^i - x_E^i)^2 + (\overline{y}_E^i - y_E^i)^2\right] \quad p \in \mathbb{R}^{29}. \tag{12}$$

4.2. Restricciones

Las restricciones de diseño se expresan con las ecuaciones (13) correspondientes:

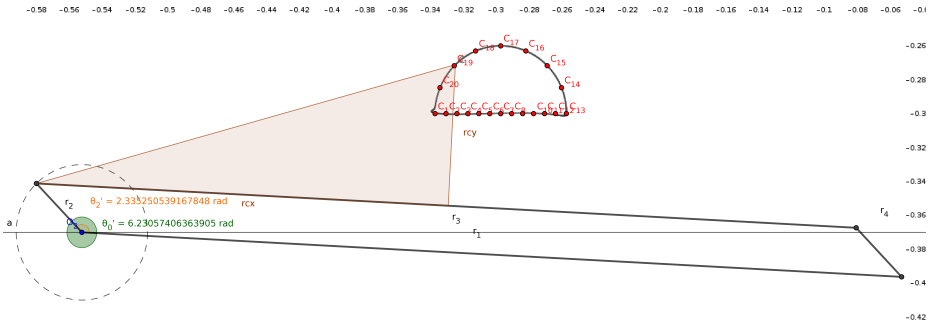


Fig. 3. Mecanismo óptimo: Evolución diferencial.

$$
\begin{aligned}
g_1(p) &= l_2 + l_3 - l_1 - l_4 \leq 0 \\
g_2(p) &= l_2 - l_3 \leq 0 \\
g_3(p) &= l_2 - l_4 \leq 0 \\
g_4(p) &= l_2 - l_1 \leq 0 \\
g_5(p) &= l_1 - l_3 \leq 0 \\
g_6(p) &= l_4 - l_3 \leq 0 \\
g_7(p) &= \theta_2^{10} - \theta_2^{11} \leq 0 \\
g_8(p) &= \theta_2^{11} - \theta_2^{12} \leq 0 \\
g_9(p) &= \theta_2^{12} - \theta_2^{13} \leq 0 \\
&\vdots \\
g_{25}(p) &= \theta_2^{28} - \theta_2^{29} \leq 0,
\end{aligned}
\tag{13}
$$

donde: los valores de los límites superior e inferior de las variables de diseño propuestos en [13] se observan en la Tabla 1.

5. Algoritmos de optimización

5.1. Búsqueda armónica

Algoritmo heurístico denominado HS (Harmony Search) funciona como el proceso de imitar la improvisación en una interpretación musical [10]. Cuando un músico está improvisando, éste realiza una de las siguientes acciones [3]:

1. Toca alguna melodía conocida que ha aprendido anteriormente.
2. Toca algo parecido a la melodía anteriormente mencionada, ajustándola poco a poco al tono deseado.
3. Compone una nueva melodía basándose en sus conocimientos musicales para seleccionar nuevas notas aleatoriamente.

Tabla 2. Vector de variables de diseño: Búsqueda armónica.

No.	l_1	l_2	l_3	l_4	r_cx	r_cy	θ_0	x_0	y_0	θ_2^1
1	0.473220235	0.032348431	0.499998021	0.16732344	0.190263529	0.277719669	2.623727398	0.513412554	-0.044473163	1.456826347
2	0.357548313	0.02766025	0.499997377	0.215078399	0.398151669	0.4845414	0.096656775	-0.504103062	-0.897136203	4.875237892
3	0.230598661	0.024717261	0.23555831	0.224274507	0.08147685	0.479662013	5.18714452	-0.132433173	-0.754448593	5.843051441
4	0.473785354	0.044503366	0.499950923	0.234468952	0.313458259	0.097885115	5.936458913	-0.427542003	-0.42356719	4.286482092
5	0.499976136	0.040223415	0.499990628	0.471766814	0.388987613	0.496087111	2.171576131	0.042798361	0.216308851	2.58386484
6	0.49998144	0.031916127	0.499998914	0.375044365	0.224940904	0.343949447	5.371381307	0.055924119	-0.592084039	5.091658307
7	0.310054364	0.024934815	0.499996165	0.280420971	0.361783114	0.488451561	6.193476265	0.043113645	-0.878503522	4.753038661
8	0.4837542	0.025588016	0.483763782	0.456918592	0.01755792	0.300690404	1.63542402	-0.01072323	-0.031700189	2.303140839
9	0.2965412	0.034737921	0.49729193	0.336792048	0.468449808	0.499950675	3.089174274	-0.255573413	0.406320105	1.484799349
10	0.499985135	0.030308522	0.499996356	0.353298103	0.178549775	0.369635285	5.32047189	-0.341865792	-0.599641824	5.145381557
11	0.332377987	0.030267334	0.499991727	0.398256894	0.36469443	0.461863265	2.719237038	0.169837093	0.298938252	1.685892377
12	0.497270992	0.042077503	0.49999794	0.045802669	0.287464472	0.1577466	0.009477661	0.173080739	-0.466452908	4.357711398
13	0.469612045	0.028317258	0.483601031	0.247263493	0.094346999	0.382458402	2.26227848	0.00951829	0.041765236	1.903460226
14	0.393626859	0.031227668	0.499993407	0.239235876	0.307411243	0.311213369	6.025917772	-0.675472455	-0.657442267	4.798831606
15	0.375849593	0.030099818	0.499998361	0.273443904	0.354690807	0.483041976	2.865752793	0.073499166	0.279417963	1.950012352
16	0.483090594	0.025826762	0.483112561	0.48310485	0.035473685	0.27989311	1.675524479	-0.245971882	-0.042414933	2.230121516
17	0.428323133	0.029332525	0.499994217	0.310189384	0.267991793	0.482605648	5.641375033	-0.216514115	-0.772925214	5.00337375
18	0.371102184	0.027736338	0.495439776	0.439147904	0.320373143	0.497853853	5.542765985	-0.553723073	-0.851294561	5.448851125
19	0.465130746	0.033363127	0.499994785	0.179088721	0.244353186	0.361411395	5.775649223	-0.426001761	-0.605508409	4.917821651
20	0.420306478	0.031498786	0.499996869	0.46608843	0.341338075	0.496129164	5.30743682	-0.591383695	-0.801402614	5.883065375
No.	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6	θ_2^7	θ_2^8	θ_2^9	θ_2^10	θ_2^11
1	1.607717607	1.764846423	1.934375266	2.117689894	2.262306383	2.459292729	2.625418371	2.804695009	2.978571167	3.160961162
2	4.986551868	5.084339723	5.161516226	5.269341592	5.35435459	5.425912562	5.511801533	5.605582008	5.687173312	5.760303503
3	5.944685999	6.042169371	6.129622554	6.216072086	0.020549953	0.106804296	0.18274476	0.268720541	0.364404945	0.447121306
4	4.464122331	4.634475545	4.803509498	4.952514241	5.085347146	5.227752314	5.351277061	5.47424548	5.62178953	5.751516023
5	2.69346462	2.776273301	2.877600243	2.971184255	3.065702956	3.154802705	3.253766342	3.35534978	3.445906324	3.548448558
6	5.227046836	5.364957845	5.518565952	5.687172744	5.786436907	5.96744162	6.090860924	6.23093001	0.107998273	0.2772556
7	4.884295421	4.987031505	5.115591588	5.195468915	5.289281561	5.387204733	5.478547842	5.556862753	5.648263794	5.738904836
8	2.428864774	2.585007506	2.743755919	2.930284764	3.120581137	3.30629435	3.47309067	3.673021926	3.885153044	4.076182355
9	1.59314161	1.694740506	1.79267986	1.882158671	1.964565214	2.047771093	2.126656724	2.204129584	2.280691235	2.352518368
10	5.297438174	5.455774426	5.611597744	5.76347223	5.879626523	6.054869053	6.179460423	0.052758811	0.180319957	0.380751179
11	1.855386231	1.991431885	2.106878151	2.241285338	2.354136077	2.46804048	2.567727743	2.667035782	2.768793422	2.869509054
12	4.576514075	4.743145545	4.895886088	5.032665145	5.165114568	5.294098039	5.416215053	5.546107989	5.671667199	5.806669348
13	2.068673264	2.231424123	2.396993586	2.556034623	2.72246835	2.907700134	3.040381667	3.232188203	3.38641017	3.561889223
14	4.966077441	5.113288874	5.232510454	5.347799643	5.46673551	5.594361322	5.716884158	5.838792731	5.938535388	6.089385015
15	2.067652804	2.185626752	2.303194403	2.41565523	2.530228449	2.643890146	2.753969416	2.881619114	2.985051273	3.09193226
16	2.389292194	2.512769504	2.695734335	2.865828736	3.047446126	3.26102275	3.433428978	3.615914036	3.826574703	4.025430253
17	5.157258757	5.304305427	5.434447373	5.567591294	5.683668606	5.79770865	5.922697146	6.023900029	6.13543603	6.26518001
18	5.569968813	5.690830788	5.805911937	5.91084514	6.017814681	6.11654951	6.215508349	0.048755821	0.149436277	0.243450331
19	5.067424773	5.202492345	5.329799988	5.454606196	5.577412211	5.709979159	5.838178798	5.96778277	6.11598348	6.254414149
20	5.986167038	6.084631884	6.191265897	0.0029103	0.087329375	0.183114279	0.292198116	0.394863967	0.51745166	0.634254361
No.	θ_2^12	θ_2^13	θ_2^14	θ_2^15	θ_2^16	θ_2^17	θ_2^18	θ_2^19	θ_2^20	Función Objetivo
1	3.335781771	3.486841887	3.927478995	4.838980492	5.213231296	5.541750685	5.890272273	6.261178115	1.186550948	0.000318649
2	5.844441163	5.945461111	5.974937986	1.763280794	1.978290209	2.208595293	2.446057296	2.665815987	2.88654334	0.000449399
3	0.532398246	0.623617969	2.881959154	3.046977565	3.250978666	3.468838317	3.676414935	3.895216531	4.066062345	0.000494653
4	5.897571324	6.066010957	1.339334944	1.690269016	1.990362929	2.295872132	2.604776449	2.965948331	3.627312267	0.000335221
5	3.657016899	3.780831998	5.701217617	5.865886022	6.057524203	6.246967193	0.164156035	0.304316804	2.563492759	0.000454075
6	0.420992789	0.602731182	1.877192403	2.323849257	2.636722119	2.963046178	3.316645437	3.692831916	4.769116231	0.000332342
7	5.845452999	5.932803689	5.940443209	2.075918302	2.282285078	2.515777115	2.755659328	3.023189366	3.256626338	0.000424489
8	4.285577961	4.44021464	5.001391539	5.601911694	6.026115404	0.107624679	0.506419941	1.034690063	2.041276467	0.000389175
9	2.426879021	2.49773746	2.499991182	5.871332208	5.967109147	6.09643141	6.254658366	0.12447535	0.280598801	0.000449118
10	0.516801166	0.725521526	1.740227059	2.210962619	2.604149765	2.942668786	3.296516239	3.674242569	4.900952057	0.00028713
11	2.979628908	3.070947511	3.098652414	5.869717709	6.010946988	6.198225295	0.135276308	0.36274376	0.624513541	0.000437195
12	5.955595615	6.128386729	0.525950669	0.93680096	1.296347928	1.636128364	1.9778168	2.353431238	2.743588304	3.25E-005
13	3.748972959	3.898167233	4.347359234	5.154114863	5.566630632	5.901139351	6.230640599	0.310677622	1.668034722	0.000309168
14	6.244530605	0.102714322	1.302683195	1.682548968	1.991435908	2.300758058	2.591074806	2.889967215	3.226976407	0.000340147
15	3.222122675	3.347685378	3.389658259	5.161915198	5.36771907	5.594794634	5.812898345	6.002521754	6.169678821	0.000506715
16	4.239728075	4.37272447	4.92373341	5.592834512	5.970091933	0.146173088	0.519805045	1.032006302	1.85179119	0.000369421
17	0.110236925	0.248501174	2.071020859	2.355742532	2.617084847	2.889587882	3.187567151	3.496900865	3.90165463	0.000421877
18	0.345516265	0.48221924	0.491719301	2.513012776	2.752433556	3.00935004	3.259689758	3.525217444	3.77513217	0.000387777
19	0.14556231	0.320044966	1.314598353	1.758934009	2.108647541	2.443175883	2.752519609	3.086523444	4.837433012	0.00028347
20	0.774771572	0.941675822	2.049684522	2.355940279	2.622298255	2.880215059	3.110587596	3.318440956	5.869616363	0.00037408

Descripción del algoritmo HS El algoritmo HS se relaciona a un vector de variables de diseño $X^j = (x_1, x_2, \cdots, x_N)$, con armonías, llamado vector armónico donde $j = 1, 2, \cdots, k$.

– **Paso 1. Inicializa una memoria armónica (MA)**. La memoria armónica (MA) $X = (x_1, x_2, ..., x_N)$ se genera con una distribución uniforme considerando $L_i < x_i^j < U_i$, donde L_i y U_i son los límites inferior y superior respectivamente del problema dado, con $i = 1, 2, \cdots, N$ resultando (14).

$$\text{MA} = \begin{bmatrix} X^1 \\ X^2 \\ \vdots \\ X^k \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_N^1 \\ x_1^2 & x_2^2 & \cdots & x_N^2 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^k & x_2^k & \cdots & x_N^k \end{bmatrix}, \tag{14}$$

– **Paso 2. Improvisa una nueva armonía**. Se genera un nuevo vector armónico $X_{new} = (x_1^{new}, x_2^{new}, x_3^{new}, ..., x_N^{new})$ estimando dos probabilidades, $r_{accept} \in [0, 1]$ y $r_{pa} \in [0, 1]$. Con probabilidad r_{accept} se elige de manera aleatoria un índice j de la memoria armónica, y con probabilidad $(1 - r_{accept})$ se elige $x_i^{new} = rand$, que es lo que se interpreta como improvisación.

– **Paso 3.Ajuste de armonía**. Con probabilidad $r_{accept} r_{pa}$ se asigna $x_i^{new} = x_i^k + (rand)(bw)$ donde $x_i^k \in MA$, $rand \in [0, 1]$ y bw es un ancho de banda que define el rango de ajuste. Si no se cumple, es decir, con probabilidad $1 - r_{accept} r_{pa}$ se asigna $x_i^{new} = MA(j, i)$ (j es el índice elegido en Paso 2).

– **Paso 4.Actualizar memoria armónica**. Si la nueva armonía X_{new} produce un mejor desempeño que la peor armonía de la MA entonces se sustituye actualizándola por la mejor.

– **Paso 5.Si el criterio de paro no se cumple, ir al paso 2**. Terminar cuando el número máximo de iteraciones o ciclos se alcanza.

Los Pasos 1 a 5 se plasman en el pseudocódigo de HS que se muestra en el Algoritmo 1.1 [22].

5.2. Evolución Diferencial

Rainer Storn y Kenneth Price [16] desarrollaron un algoritmo denominado Evolución Diferencial (ED), este algoritmo heurístico es un optimizador basado en la población que ataca el problema de punto de partida al muestrear la función objetivo en múltiples puntos iniciales elegidos al azar [14]. Las fases principales de este método son:

– **Inicialización.** Antes de iniciar la población, los limites inferior b_L y superior b_U deberán ser definidos, los valores de los parámetros de cada vector son generados aleatoriamente dentro del rango determinado por los límites, la ecuación (15) representa el vector resultante:

$$x_{j,i} = b_{j,L} + (b_{j,U} - b_{j,L}) * rand(0, 1), \tag{15}$$

Tabla 3. Vector de variables de diseño: Evolución diferencial.

No.	l_1	l_2	l_3	l_4	r_cx	r_cy	θ_0	x_0	y_0	θ_2^1
1	0.034142014	0.018956686	0.244068043	0.244066686	1.12E-005	0.041842632	4.739602097	-0.02425582	-0.321903669	5.382583148
2	0.035034132	0.019253992	0.274117666	0.272961319	0.007617314	0.04163331	4.899402154	0.091365231	-0.322295495	5.289971734
3	0.499999714	0.039985917	0.499999856	0.040072264	0.251180883	0.083434351	6.230574064	-0.552043577	-0.370109899	3.833220462
4	0.499953067	0.031717969	0.499971043	0.222800686	0.152174955	0.311277952	2.434437642	0.004610382	-0.022015981	1.601552279
5	0.499978677	0.023913189	0.499983988	0.499875478	1.70E-005	0.405176328	1.725229419	0.227884289	0.094072265	2.296687525
6	0.49999255	0.040684817	0.499995741	0.464837779	0.244670596	9.08E-006	2.502653207	-0.111789446	-0.204735548	1.384106147
7	0.368515424	0.024765042	0.439357141	0.407845983	0.060759876	0.265226434	5.052813156	-0.039845695	-0.532671818	5.13566821
8	0.499998502	0.024314165	0.499999078	0.456900937	3.49E-007	0.40873178	1.806454021	0.158854924	0.094855197	2.220446799
9	0.034105487	0.018914748	0.49995119	0.499950877	0.00085192	0.041957769	4.739493295	-0.299491132	-0.322072647	5.3635793
10	0.322063303	0.033613983	0.487821043	0.199531914	0.139716962	0.001743678	0.407298792	0.348304743	-0.372413513	3.528064717
11	0.235945839	0.035519065	0.318050349	0.143897238	0.118333897	0.013858241	6.263887649	-0.025322357	-0.343447153	3.949324787
12	0.499999833	0.024421757	0.499999903	0.445291904	2.04E-007	0.409583712	1.827027701	-0.140052662	0.094790498	2.201851158
13	0.49999246	0.040449377	0.499992467	0.040449389	0.254599188	0.007361777	6.233123544	-0.182546532	-0.295255027	3.393875969
14	0.046155456	0.02256725	0.148476876	0.14633798	0.027645374	0.037647777	5.23041022	-0.226567564	-0.325228906	5.140294024
15	0.03416251	0.019012473	0.195128149	0.195127121	3.35E-007	0.041639174	4.745959456	0.352541639	-0.321662589	5.387867272
16	0.036821205	0.019146952	0.142154894	0.131214552	3.09E-008	0.043362188	5.108502653	-0.115634122	-0.323446885	5.057664382
17	0.034108297	0.018910508	0.499991787	0.499247066	0.000503445	0.041991901	4.758910777	0.415502362	-0.322108067	5.340671486
18	0.034810702	0.019289841	0.243698431	0.241434758	0.007720081	0.041174542	4.941812098	-0.311237174	-0.321814721	5.250258414
19	0.388224787	0.039540018	0.499723838	0.251127152	0.223948615	0.000397852	3.118042566	-0.026365804	-0.177493457	0.887434623
20	0.499993052	0.023992409	0.499998248	0.492083745	3.27E-006	0.405498331	1.742728438	0.124460426	0.094308566	2.278124803

No.	θ_2^2	θ_2^3	θ_2^4	θ_2^5	θ_2^6	θ_2^7	θ_2^8	θ_2^9	$\theta_2^{1}0$	$\theta_2^{1}1$
1	5.640197718	5.851147784	5.991150465	6.103158596	6.203224734	0.014961	0.110348896	0.212100507	0.328049867	0.47399157
2	5.539483856	5.766956905	5.916527346	6.033254089	6.137818974	6.234532501	0.047181829	0.147219364	0.258111995	0.396113039
3	4.185234302	4.421799229	4.617049849	4.790975919	4.953012095	5.10873079	5.262521279	5.418735513	5.582105245	5.760065211
4	1.758116319	1.928843477	2.105021364	2.277704315	2.45280056	2.623996403	2.795340315	2.966245946	3.137428128	3.311076114
5	2.455548119	2.626666308	2.799846283	2.972115857	3.141636502	3.310857938	3.480255964	3.649599261	3.820626639	3.994651935
6	1.535529799	1.698158642	1.862220061	2.027337835	2.193073906	2.359350685	2.52648288	2.697773508	2.870369375	3.044419929
7	5.381353287	5.534626327	5.718653512	5.907749386	6.033883845	6.224882878	0.11715169	0.309926988	0.454854601	0.645566709
8	2.378901513	2.550328894	2.724604673	2.897522863	3.068069291	3.237249032	3.406268239	3.576020808	3.747669108	3.920459692
9	5.620276772	5.835200372	5.977685074	6.091206152	6.191577612	0.003760792	0.099122288	0.200072085	0.314359564	0.458461825
10	3.691254204	3.921719905	4.10261746	4.306832065	4.478011023	4.65916937	4.826120389	4.994169648	5.152406774	5.299278901
11	4.09611077	4.259166508	4.530483762	4.748559787	4.846437684	5.038752531	5.154767781	5.310792683	5.631949696	5.769781952
12	2.360107168	2.531534929	2.705762236	2.878685506	3.049401208	3.218667627	3.387685414	3.557601698	3.729245476	3.902150457
13	3.8281786	4.089447856	4.300957541	4.488419186	4.660938501	4.827252193	4.990832209	5.157644636	5.332479421	5.525669436
14	5.333868992	5.607060788	5.745506369	5.906958485	6.012576368	6.141882983	6.2399837	0.077939764	0.189984277	0.312387246
15	5.647822264	5.854540893	5.99455367	6.107772512	6.207014004	0.018219994	0.114614331	0.215824181	0.332100138	0.478557465
16	5.306911319	5.539892459	5.698203899	5.821691737	5.92807827	6.025766141	6.120056469	6.215782896	0.03650184	0.162327524
17	5.596945022	5.813434062	5.957158731	6.071342741	6.172200186	6.267636667	0.079746832	0.180223331	0.293733416	0.436596319
18	5.504570049	5.732870608	5.887361949	6.004502353	6.108103172	6.205539212	0.015660749	0.114759546	0.22480919	0.362605237
19	1.026704518	1.173252345	1.334264093	1.490645599	1.654929747	1.809212495	1.980886517	2.147998093	2.32018889	2.483438777
20	2.436219528	2.607985224	2.782113716	2.954355657	3.125088647	3.294052093	3.462639452	3.632820241	3.804135771	3.977013101

donde rand$(0,\ 1)$ retorna un número aleatorio distribuido uniformemente dentro del rango $[0, 1)$ y j indica un nuevo valor aleatorio para cada parámetro.

– **Mutación:** ED muta y recombina la población para producir un solo vector hijo (trial), eligiendo vectores al azar $(X_{r_0,g}, X_{r_1,g}, X_{r_2,g})$, indicando el factor escala $F \in (0, 1)$ para obtener un vector de mutación (16).

$$V_{i,g} = X_{r_0,g} + F * (X_{r_1,g} - X_{r_2,g}). \tag{16}$$

– **Cruza:** Se construye un vector hijo $U_{i,g}$ (17) entre el vector padre $X_{i,g}$ y el vector de mutación $V_{i,g}$.

$$U_{i,g} = u_{j,i,g} \begin{cases} V_{j,i,g} & \text{si}(\text{rand}_j(0,1)) \leq Cr \quad \text{o} \quad j = j_{\text{rand}}, \\ X_{j,i,g} & \text{en otro caso.} \end{cases} \tag{17}$$

El factor de cruza $Cr \in [0, 1]$ es una probabilidad predefinida, la cual controla la fracción de parámetros que se heredan. Para garantizar que el hijo no sea un duplicado del padre se elige un número aleatorio, si éste es menor que Cr, el parámetro se hereda del vector de mutación $V_{i,g}$, en cambio éste es heredado por el padre $X_{i,g}$.

– **Selección:** El operador de selección decide que individuo es aceptado (18), si el hijo $U_{i,g}$ tiene un desempeño igual o inferior al padre $X_{i,g}$, este último es reemplazado, de lo contrario el padre conserva su lugar, ambos por al menos una generación más.

$$X_{i,g+1} = \begin{cases} U_{i,g} \text{ si } \quad f(U_{i,g}) \leq f(X_{i,g}), \\ X_{i,g} \text{ en otro caso.} \end{cases} \tag{18}$$

Producida la nueva población, el procedimiento de mutación, recombinación y selección se vuelve iterativo hasta localizar el óptimo o cumplir el máximo de generaciones definidas por el usuario. El pseudocódigo de ED se muestra en el Algoritmo 1.2 [16].

6. Resultados

6.1. Desempeño de Búsqueda Armónica

Los resultados obtenidos se muestran en la Tabla 2, éstos resultados corresponden a los 20 mejores vectores de las variables de diseño alcanzados con el algoritmo de Búsqueda Armónica. Con el propósito de visualizar el valor correspondiente de la función objetivo, a la Tabla 2, se le ha agregado el valor de desempeño para cada vector de diseño representado en un campo como **Función Objetivo**. El vector de diseño resultante que contiene la mejor solución (mínimo valor obtenido de la función objetivo) se resalta en color amarillo. El vector solución con mejor desempeño por HS cuyo valor óptimo en la función objetivo corresponde a:

$$\begin{aligned} p = \{&0{,}497270992, 0{,}042077503, 0{,}49999794, 0{,}045802669, 0{,}287464472, \\ &0{,}1577466, 0{,}009477661, 0{,}173080739, -0{,}466452908, 4{,}357711398, \\ &4{,}576514075, 4{,}743145545, 4{,}895886088, 5{,}032665145, 5{,}165114568, \\ &5{,}294098039, 5{,}416215053, 5{,}546107989, 5{,}671667199, 5{,}806669348, \\ &5{,}955595615, 6{,}128386729, 0{,}525950669, 0{,}93680096, 1{,}296347928, \\ &1{,}636128364, 1{,}9778168, 2{,}353431238, 2{,}743588304\}, \end{aligned} \tag{19}$$

$$\text{mín}(F.O._{HS}) = 3{,}24685 \times 10^{-05}. \tag{20}$$

A partir del óptimo obtenido y su vector correspondiente (ecuación 20 y 19) se realiza la simulación mostrada en la Figura 2.

6.2. Desempeño evolución diferencial

Las Tablas 3 y su continuación que es la Tabla 4 indican los resultados alcanzados con el algoritmo de Evolución Diferencial, se puede visualizar el valor óptimo

Tabla 4. Dimensión óptima de las 4 barras del mecanismo

Algoritmo	l_1	l_2	l_3	l_4	r_cx	r_cy
Búsqueda Armónica	0.497270992	0.042077503	0.49999794	0.045802669	0.287464472	0.1577466
Evolución Diferencial	0.499999714	0.039985917	0.499999856	0.040072264	0.251180883	0.083434351

Algoritmo	θ_0	x_0	y_0	θ_2^1	θ_2^2	θ_2^3
Búsqueda Armónica	0.009477661	0.173080739	-0.466452908	4.357711398	4.576514075	4.743145545
Evolución Diferencial	6.230574064	-0.552043577	-0.370109899	3.833220462	4.185234302	4.421799229

Algoritmo	θ_2^4	θ_2^5	θ_2^6	θ_2^7	θ_2^8	θ_2^9
Búsqueda Armónica	4.895886088	5.032665145	5.165114568	5.294098039	5.416215053	5.546107989
Evolución Diferencial	4.617049849	4.790975919	4.953012095	5.10873079	5.262521279	5.418735513

Algoritmo	θ_2^10	θ_2^11	θ_2^12	θ_2^13	θ_2^14	θ_2^15
Búsqueda Armónica	5.671667199	5.806669348	5.955595615	6.128386729	0.525950669	0.93680096
Evolución Diferencial	5.582105245	5.760065211	5.968818105	0.03176862	0.443501996	0.838253583

Algoritmo	θ_2^16	θ_2^17	θ_2^18	θ_2^19	θ_2^20	**Función Objetivo**
Búsqueda Armónica	1.296347928	1.636128364	1.9778168	2.353431238	2.743588304	3.24685E-05
Evolución Diferencial	1.232165409	1.62540309	2.018496404	2.41125571	2.805075923	1.29E-06

alcanzado en la fila 3, donde la Función Objetivo es $1{,}29E^{-6}$. El valor óptimo que se obtuvo (22) y el vector con las 29 variables de diseño dado en (21) marcado en las Tablas 3 y 4 de color amarillo, obedece a:

$$\begin{aligned} p = \{&0{,}499999714, 0{,}039985917, 0{,}499999856, 0{,}040072264, 0{,}251180883, \\ &0{,}083434351, 6{,}230574064, -0{,}552043577, -0{,}370109899, 3{,}833220462, \\ &4{,}185234302, 4{,}421799229, 4{,}617049849, 4{,}790975919, 4{,}953012095, \\ &5{,}10873079, 5{,}262521279, 5{,}418735513, 5{,}582105245, 5{,}760065211, \\ &5{,}968818105, 0{,}03176862, 0{,}443501996, 0{,}838253583, 1{,}232165409, \\ &1{,}62540309, 2{,}018496404, 2{,}41125571, 2{,}805075923\}, \end{aligned} \tag{21}$$

$$min(F.O._{ED}) = 1{,}29x10^{-06}. \tag{22}$$

La simulación del caso de estudio del óptimo obtenido con ED se muestra en la Figura 3. En resumen la Tabla 4 muestra los valores del ambos sistemas óptimos obtenidos resultado de la implementación de cada algoritmo. Como se puede apreciar el valor óptimo fue obtenido con Evolución diferencial dado por $1{,}29E-06$. Cabe mencionar que las muestras presentadas en la Tabla 2, para Búsqueda Armónica solo el $5\,\%$ cumple con condición de que una solución factible satisfaga la tolerancia $\delta = 1E-04$, mientras que para Evolución Diferencial (Tablas 3 y 4) éste porcentaje es de $50\,\%$.

7. Conclusión

El algoritmo de Evolución Diferencial para resolver problemas de optimización con restricciones produjo un mejor desempeño comparado con el algoritmo de Búsqueda Armónica. La síntesis dimensional del mecanismo de cuatro barras óptimo, que se genera a partir de los resultados, podemos calificarlo como satisfactorio debido a que, como se observa en la Figura 3 genera una trayectoria casi análoga del semicírculo planteado como caso de estudio.

Referencias

1. Askarzadeh, A., Rezazadeh, A.: Parameter identification for solar cell models using harmony search-based algorithms. Solar Energy, vol. 86, no. 11, pp. 3241–3249 (2017) doi: 10.1016/j.solener.2012.08.018
2. Bäck, T.: Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms. Oxford University Press (1996)
3. Cobos, C., Pérez, J., Estupiñan, D.: Una revisión de la búsqueda armónica. Revista Avances en sistemas e Informática, vol. 8, no. 2, pp. 67-80 (2011)
4. Deb, K.: An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanics and Engineering, vol. 186, pp. 311–338 (2000) doi: 10.1016/S0045-7825(99)00389-8
5. Echevarría, L. C. and Orestes, S., and da-Silva, A. J.: Aplicación de los algoritmos evolución diferencial y colisión de partículas al diagnóstico de fallos en sistemas industriales. Revista Investigación Operacional, vol. 33, no. 2, pp. 160–172 (2012)
6. Foix, I., Cardona, S., Costa, D. C.: Teoría de máquinas. Universidad Politècnica de Catalunya, vol. 95 (2001)
7. Garro, B., Sossa, H., Vazquez, R. A.: Diseño automático de redes neuronales artificiales mediante el uso del algoritmo de evolución diferencial (ED). Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, vol. 46, pp 13–27 (2012)
8. Guzmán-Gaspar, J. Y.: Evolución diferencial para el control de un motor de corriente directa bajo incertidumbre paramétrica. Tesis de maestría, Centro de enseñanza LANIA (2015)
9. Geem, Z., Choi, J. Y.: Music composition using harmony search algorithm. Workshops on Applications of Evolutionary Computation, pp. 593–600 (2007) doi:10.1007/978-3-540-71805-5_65
10. Geem, Z.,Kim, J. and Loganathan, G.: A New heuristic optimization algorithm: Harmony search simulation. Applied Soft Computing, vol. 76, pp. 60–68 (2001) doi: 10.1177/003754970107600201
11. Geem, Z.: Optimal cost design of water distribution networks using harmony search. Engineering Optimization, vol. 38, no. 03, pp. 259–277 (2006) doi: 10.1080/03052150500467430
12. Myszka, D. H.: Máquinas y mecanismos. Pearson, 4ta edición (2012)
13. Pantoja-García, J.S., Villarreal-Cervantes, M.G., González-Robles, J.C., Cervantes, G. S., Síntesis óptima de un mecanismo para la marcha bípeda utilizando evolución diferencial, Elsevier, Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería, vol. 33, no. 1, pp 138–153 (2017)
14. Price, K., Storn, R., Lampinen, M., Jouni, A.: Differential evolution: A practical approach to global optimization. Springer Science and Business Media (2006)

15. Saka, M. P.: Optimum geometry design of geodesic domes using harmony search algorithm. Advances in Structural Engineering, vol. 10, no. 6, pp. 595–606 (2007) doi: 10.1260/136943307783571445
16. Storn, R. and Price, K., Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces. International Computer Science Institute, vol. 11, pp. 341–359 (1995)
17. Vázquez, K. R.: Ant colony optimization. Genetic Programming and Evolvable Machines, vol. 6 (2005) doi: 10.1007/s10710-005-2991-z
18. Vega-Alvarado, E., Santiago-Valentín, E., Sánchez-Márquez, A., Solano-Palma, A., Portilla-Flores, E. A., Flores-Pulido, L., Síntesis óptima de un mecanismo plano para seguimiento de trayectoria utilizando evolución diferencial. Research in Computing Science, Vol. 72, pp. 85–98 (2014)
19. Villate-Gil, A., Rincón-Arandia, D. E., Melgarejo-Rey, M. A.: Evolución diferencial aplicada a la sintonización de clasificadores difusos para el reconocimiento del lenguaje de señas. Ingeniería y Universidad, vol. 16, no. 2, pp. 397–413 (2012)
20. Yang, X. S.: Engineering optimizations via nature-inspired virtual bee algorithms. Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach. IWINAC 2005. Lecture Notes in Computer Science, vol. 3562, pp. 317–323 (2005) doi: 10.1007/11499305_33, pp. 317-323
21. Yang, X.: Firefly algorithms for multimodal optimization. International symposium on stochastic algorithms, pp. 169–178 (2009) doi: 10.1007/978-3-642-04944-6_14
22. Yang, X.: Harmony search as a metaheuristic algorithm. Music-inspired harmony search algorithm: theory and applications, pp. 1–14 (2009) doi: 10.1007/978-3-642-00185-7_1

ISSN 1870-4069

Selección de características y optimización de hiperparámetros para la mejora en la clasificación del cáncer de próstata

Andrea G. Plascencia-Rodríguez[1], Manuel A. Soto-Murillo[1], José M. Celaya-Padilla[1], Jorge I. Galván-Tejada[1], Carlos E. Galván-Tejada[1]

Universidad Autónoma de Zacatecas,
Unidad Académica de Ingeniería Eléctrica,
Zacatecas,
México

{andrea.plascencia, 28900587, jose.celaya ,gatejo, ericgalvan}@uaz.edu.mx

Resumen. La selección precisa de características en el análisis de expresión génica es crucial para comprender la biología subyacente y mejorar el diagnóstico y tratamiento del cáncer de próstata. En este estudio, SE aplica la técnica de suma de cuadrados entre grupos y dentro de grupos (BSS/WSS) para identificar genes relevantes en la clasificación de muestras tumorales y normales. Los resultados muestran que la selección de características mejoró la eficiencia de los modelos de Árboles de Decisión y Bosques Aleatorios, alcanzando una precisión prometedora en la clasificación de muestras de cáncer de próstata. La optimización de hiperparámetros, especialmente en los modelos de Bosques Aleatorios, demostró un rendimiento óptimo. Estos hallazgos resaltan la importancia de la selección de características en la investigación del cáncer de próstata y sugieren su relevancia clínica para la práctica médica y la salud pública.

Palabras clave: Selección de características, expresión génica, cáncer de próstata, suma de cuadrados entre grupos y dentro de grupos, regresión logística, árboles de decisión, bosques aleatorios.

Feature Selection and Hyperparameter Optimization for Improved Prostate Cancer Classification

Abstract. Precise feature selection in gene expression analysis is crucial for understanding underlying biology and improving the diagnosis and treatment of prostate cancer. In this study, we apply the between-group sum of squares and within-group sum of squares (BSS/WSS) technique to identify relevant genes in the classification of tumor and normal samples. The results show that feature selection enhanced the efficiency of decision trees and random forest models, achieving promising accuracy in prostate cancer sample classification. Hyperparameter optimization, especially in random forest models, demonstrated

optimal performance. These findings underscore the importance of feature selection in prostate cancer research and suggest its clinical relevance for medical practice and public health.

Keywords: feature selection, gene expresión, prostate cancer, between-group sum of squares and within-group sum of squares, logistic regre-ssion, decision tree, random forests.

1. Introducción

El cáncer es una enfermedad causada por alteraciones genómicas en el ADN, el ARN y las proteínas de una célula, que conducen a un crecimiento y desa-rrollo celular anormal. Comprender estas alteraciones genómicas es crucial para decodificar los mecanismos del desarrollo del cáncer y mejorar el diagnóstico y tratamiento de los diferentes tipos de cáncer en función de sus anomalías moleculares [1]. En el contexto del cáncer de próstata, la progresión hacia la malignidad de la próstata se caracteriza por una serie secuencial de pasos.

Estas etapas comienzan con el desarrollo de la neoplasia intraepitelial prostática (PIN), a la que sigue la aparición de un cáncer de próstata localizado. Posteriormente, aparece una forma avanzada de adenocarcinoma de próstata con invasión local que, en última instancia, culmina en un cáncer de próstata metastásico [12]. El cáncer de próstata, una neoplasia que afecta a la glándula prostática, se considera uno de los problemas de salud más importantes entre la población masculina.

Representa una carga importante a escala mundial, ya que se sitúa como la segunda causa principal de cáncer y la quinta causa principal de mortalidad relacionada con el cáncer en los hombres [6]. Esta dolencia presenta una amplia gama de perfiles moleculares y heterogeneidades genéticas, lo que complica su diagnóstico y la implementación de estrategias de tratamiento eficaces [11].

La detección temprana del cáncer de próstata es imperativo para mejorar las tasas de supervivencia y la calidad de vida de los pacientes. Aunque se han establecido métodos convencionales como el antígeno prostático específico (PSA) y la biopsia, estos enfrentan desafíos significativos en términos de especificidad y sensibilidad, lo que subraya la necesidad apremiante de enfoques más precisos y no invasivos [7]. En este contexto, la aplicación de técnicas de aprendizaje automático ha surgido como un enfoque innovador y prometedor para abordar los desafíos asociados con la detección y clasificación del cáncer de próstata.

La capacidad del aprendizaje automático para analizar grandes conjuntos de datos ómicos y otros recursos proporciona una oportunidad única para identificar patrones moleculares y biomarcadores asociados con la enfermedad [10]. Estos elementos clave resultan fundamentales para esclarecer la complejidad de la expresión génica, un principio fundamental en la biología molecular. El principio fundamental de la biología molecular postula que el ácido deso-xirribonucleico (ADN) engendra ácido ribonucleico (ARN) y el ARN engendra proteínas. Este fenómeno se conoce como expresión génica, en el que la información genética se utiliza dentro de una entidad celular para generar las proteínas necesarias para la funcionalidad celular.

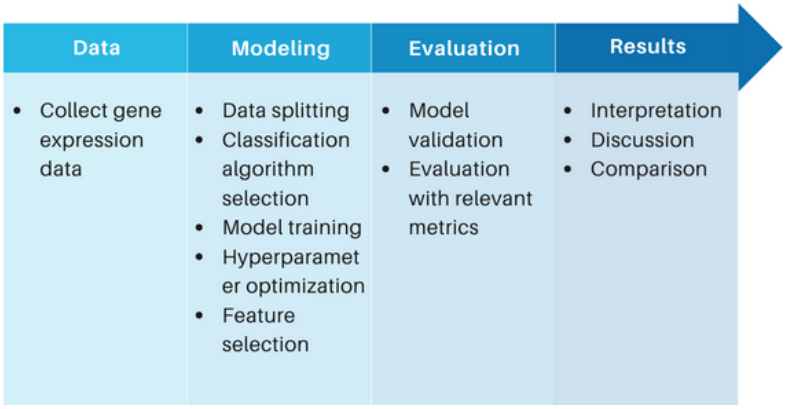


Fig. 1. Flujo del proceso.

Más específicamente, el modelo para la síntesis de proteínas está integrado en la secuencia de nucleótidos del ADN. La transformación de la información codificada por los genes en proteínas constituye un proceso celular que depende de los ácidos nucleicos. La expresión génica abarca dos procedimientos complejos: la transcripción y la traducción. La transcripción denota el acto de sintetizar ARN a partir del ADN, iniciando así la cascada de expresión génica y sirviendo como punto de control crítico en la síntesis de proteínas y la manifestación génica [9].

La investigación en el campo de la expresión génica ha revolucionado nuestra comprensión de las enfermedades. La abundancia de datos de expresión génica proporciona una ventana única hacia los complejos mecanismos moleculares subyacentes a esta enfermedad, pero también presenta desafíos significativos debido a la alta dimensionalidad y variabilidad inherentes a estos conjuntos de datos.

Dentro de este ámbito, la selección de características emerge como un componente esencial para destilar la información más relevante y biológicamente significativa de los datos de expresión génica del cáncer de próstata. La identificación de biomarcadores específicos y la comprensión de las firmas genéticas distintivas pueden desempeñar un papel crucial en el diagnóstico temprano, la estratificación de pacientes y el desarrollo de terapias más personalizadas.

En esta revisión se explora la selección de características aplicadas a datos de expresión génica, con un enfoque especial en su aplicación al estudio del cáncer de próstata. Se examina la evolución de estas técnicas, desde enfoques clásicos hasta enfoques más avanzados, destacando sus aplicaciones, limitaciones y contribuciones específicas a la comprensión de la biología subyacente al cáncer de próstata.

2. Materiales y métodos

La metodología empleada en este estudio sigue un enfoque sistemático y organizado para llevar a cabo la investigación como se muestra en la Fig. 1 Comienza con la recopilación de datos de expresión génica, esenciales para el análisis y la comprensión del problema en estudio.

Tabla 1. Impacto de la profundidad máxima en el rendimiento del árbol de decisión.

Profundidad máxima	AUC
1	0.735
2	0.619
3	0.629
10	0.705
20	0.705
30	0.685
40	0.655
50	0.695
100	0.690
None	0.705

Posteriormente, se procede con el modelado de los datos, donde se dividen en conjuntos de entrenamiento, prueba, y validación, se seleccionan los algoritmo de clasificación adecuados, se entrenan los modelos con los datos de entrenamiento y se optimizan los hiperparámetros para mejorar su rendimiento. Luego, se evalúan los modelos resultantes utilizando métricas pertinentes para validar su precisión y efectividad. En la etapa de resultados, se interpreta el significado de los hallazgos.

Posteriormente, se procede con el proceso de selección de características, donde se identifica un subconjunto óptimo de características a partir del conjunto original. Este proceso de selección de características es fundamental para destilar la información más relevante y biológicamente significativa de los datos de expresión génica del cáncer de próstata. Este nuevo subconjunto se utiliza para comenzar de nuevo desde el modelado. En conjunto, esta metodología proporciona una guía clara y estructurada para el proceso de investigación, desde la recolección de datos hasta la interpretación y discusión de los resultados, asegurando la coherencia y la rigurosidad en cada etapa del estudio.

3. Selección de características

El objetivo de la selección de características es identificar un subconjunto óptimo, representado por M **características**, a partir del conjunto original de N **dimensiones** (donde $M \leq N$), con el fin de maximizar la función objetivo. Dado un conjunto de características $X = \{x_i, i = 1, \ldots, N\}$, se busca un subconjunto $Y_M = \{x_{1i}, x_{2i}, \ldots, x_{iM}\}$ con $M \leq N$, que optimice la función objetivo $J(Y)$, la cual está relacionada con la probabilidad de clasificación correcta de alguna manera. La función objetivo, que evalúa la calidad del subconjunto de características, puede vincularse con la precisión predictiva, en el caso del enfoque del wrapper, o calcularse a partir del contenido de información del propio subconjunto (por ejemplo, distancia entre clases, correlación o medidas teóricas de información), siguiendo el enfoque de los filtros. La selección de características permite que las características seleccionadas mantengan su interpretación física original, lo cual facilita la comprensión del proceso físico subyacente en la generación de patrones.

Tabla 2. Evaluación de parámetros del árbol de decisión.

AUC			
Profundidad máxima	**1**	**10**	**50**
Número mínimo de muestras por hoja			
1	0.735	0.705	0.650
5	0.735	0.831	0.856
10	0.735	0.826	0.826
15	0.735	0.818	0.819
20	0.735	0.670	0.692
100	0.830	0.790	0.680
200	0.500	0.500	0.500
500	0.500	0.500	0.500

Además, puede resultar en una reducción de los costos de medición y/o computacionales, ya que solo se necesitará calcular las características seleccio-nadas. Sin embargo, es importante tener en cuenta que el proceso de encontrar el mejor subconjunto de características puede ser computacionalmente exigente, y en algunos casos, podría ser necesario conformarse con una solución subóptima [2].

3.1. Suma de cuadrados entre grupos y dentro de grupos

El método de selección de características basada en la suma de cuadrados entre grupos y dentro de grupos (BSS/WSS) clasifica las características según una proporción tal que las características con una gran variación entre clases y pequeñas variaciones dentro de las clases reciben calificaciones más altas. Este algoritmo de selección de características univariado determina las características que tienen un mayor poder de discriminación entre clases [3]. Para la característica k, $x_{i,k}$ denota el valor de la característica k para el ejemplo de entrenamiento i, $\overline{x_{z,k}}$ el valor promedio de la característica k en los ejemplos de clase z, y $\overline{x_k}$ el valor promedio de la característica k en todos los ejemplos. La relación BSS/WSS del gen k la proporciona la ecuación (1), donde $\delta_{i,z}$ es igual a 1 si el ejemplo i pertenece a la clase z y 0 en caso contrario:

$$\frac{\mathrm{BSS}(k)}{\mathrm{WSS}(k)} = \frac{\sum_i \sum_z \delta_{i,z} (\overline{x_{z,k}} - \overline{x_k})^2}{\sum_i \sum_z \delta_{i,z} (x_{i,k} - \overline{x_{z,k}})^2}. \tag{1}$$

Por lo tanto, es posible ordenar las características de mayor a menor en función de la proporción BSS/WSS. Este ratio puede funcionar como un peso vinculado a una característica, ya que a mayor sea, más relevante será su capacidad para discriminar. Un interrogante relevante en este tipo de selección de características, es determinar qué grupo de características elegir. El enfoque comúnmente empleado para resolver esta cuestión es analizar la curva de precisión en función del número de características y

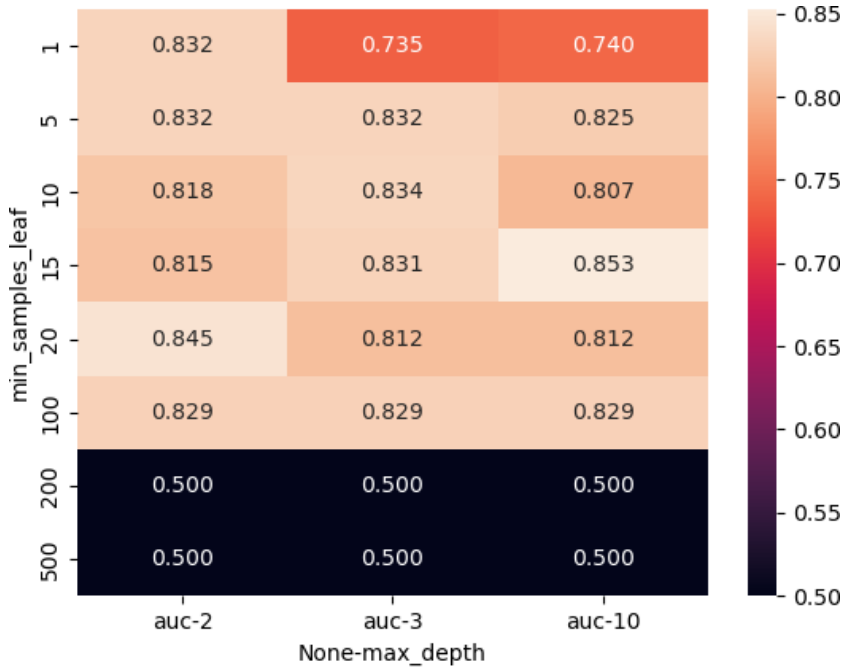


Fig. 2. Mapa de calor de puntuaciones de área bajo la curva (AUC) para árboles de decisión.

hallar un máximo local, o idealmente, un máximo global después del cual la precisión disminuye a medida que se incorporan más características. De manera más sencilla, también es factible seleccionar un número determinado de características, o limitar el conjunto de características seleccionadas en un punto de corte natural en la lista completa. Este es el enfoque utilizado en este caso.

4. Datos

La tecnología de secuenciación (RNA-seq) examina la cantidad y las secuencias de ARN en una muestra utilizando la secuenciación de próxima generación (NGS) [8]. Esta técnica analiza el transcriptoma, es decir, los patrones de expresión génica codificados dentro de nuestro ARN. En otras palabras, RNA-seq nos permite investigar y descubrir el contenido celular total de ARN, incluyendo ARNm, ARNr y ARNt. Al comprender el transcriptoma, podemos conectar la información de nuestro genoma con su expresión proteica funcional. El RNA-seq revela qué genes se activan en una célula, cuál es su nivel de expresión y cuándo se activan o desactivan [4, 5].

Por otro lado, el conjunto de datos sobre cáncer de próstata se descarga de Firehose[1]. Estos datos son el resultado de secuenciación de próxima generación, que ya ha sido normalizada. El conjunto de datos está organizado con genes en filas y pacientes en columnas.

[1] gdac.broadinstitute.org/

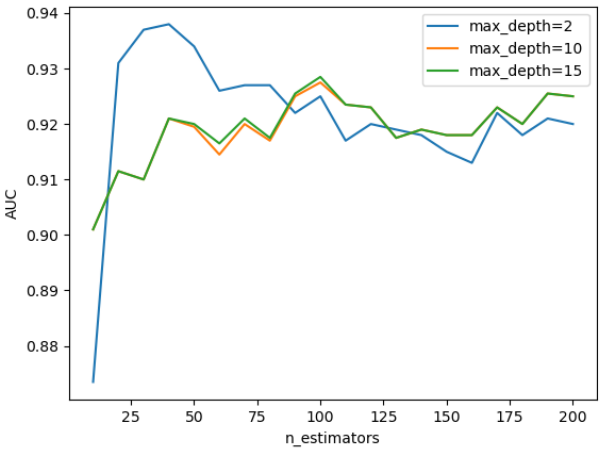


Fig. 3. Variación del AUC con respecto al número de estimadores para distintas profundidades máximas.

En total, consta de 20,531 filas que representan los genes y sus niveles de expresión, junto con 551 columnas (550 sujetos y una columna adicional para los nombres de los genes). De estas columnas, 498 corresponden a muestras tumorales y 52 a muestras normales.

5. Modelado y evaluación

Para la construcción de los modelos predictivos, primeramente se utilizó el conjunto de datos completos (550 muestras y 20,531 genes), se procedió a utilizar regresión logística, árboles de decisión y bosque aleatorio. Se dividieron los datos en conjuntos de entrenamiento (60 %), validación (20 %), y prueba (20 %) utilizando la función train_test_split de scikit-learn. Se evaluó el rendimiento de los modelos utilizando métricas relevantes como precisión y AUC (área bajo la curva) la cual resume la información de la curva ROC en una sola métrica.

Para la regresión logística, se ajustó un modelo utilizando el conjunto de entrenamiento y se evaluó su rendimiento en el conjunto de validación. Los resultados mostraron una precisión promedio del 90 % en la clasificación de muestras tumorales y normales. La matriz de confusión reveló una sensibilidad del 95 % y una especificidad del 90 %. Para los árboles de decisión, se llevó a cabo un análisis exhaustivo para determinar el impacto de la profundidad máxima del árbol de decisión en la capacidad predictiva del modelo.

Se exploraron varias profundidades, incluyendo valores específicos y sin límite de profundidad (None), y se evaluó el rendimiento del modelo utilizando el AUC. A continuación se presentan los resultados obtenidos en la Tabla 1. Los resultados muestran que la profundidad máxima del árbol de decisión influye significativamente en su capacidad para generalizar patrones en los datos de validación.

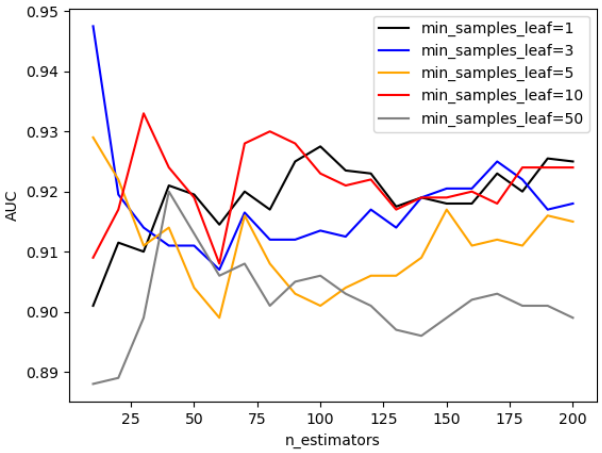


Fig. 4. Relación entre el número de estimadores y el AUC para distintos valores de min_samples_leaf.

Se observa un aumento en el AUC hasta una profundidad máxima de 10, seguido de una estabilización y, en algunos casos, un ligero descenso a profundidades mayores. Se decidió seleccionar una profundidad máxima de 10 para un equilibrio entre capacidad predictiva y complejidad del modelo.

Además, se llevaron a cabo pruebas adicionales variando tanto la profundidad máxima como el número mínimo de muestras por hoja para explorar la sensibilidad del modelo a estos parámetros. Los resultados se presentan en la Tabla 2. Estos resultados proporcionan información valiosa sobre la configuración óptima de los hiperparámetros del árbol de decisión para este conjunto de datos específico. Se realizó una evaluación comparativa entre dos modelos de árbol de Decisión para predecir la clasificación de muestras de cáncer de próstata basadas en la expresión génica.

Los modelos fueron entrenados con diferentes configuraciones de parámetros, lo que permitió comparar su rendimiento utilizando el AUC como métrica de evaluación. El primer modelo se caracterizó con una profundidad máxima de 10 y un número mínimo de muestras por hoja de 5, mientras que el segundo modelo se ajustó con una profundidad máxima de 1 y un número mínimo de muestras por hoja de 100.

Los resultados obtenidos revelan una diferencia notable en el rendimiento de los modelos. El modelo con una profundidad máxima de 1 y un número mínimo de muestras por hoja de 100 alcanzó un AUC de 0.79 en el conjunto de validación, mientras que el modelo con una profundidad máxima de 10 y un número mínimo de muestras por hoja de 5 obtuvo un AUC ligeramente inferior, con un valor de 0.76.

Este hallazgo sugiere que, a pesar de su mayor complejidad, el modelo con una profundidad máxima de 10 no logró superar significativamente al modelo más simple con una profundidad máxima de 1. La precisión comparable del modelo más simple destaca su eficacia en la tarea de clasificación de muestras de cáncer de próstata basadas en la expresión génica.

Con bosque aleatorio, de igual manera se realizó un proceso de optimización de hiperparámetros para mejorar el rendimiento en la clasificación. Se exploraron los efectos de variar los siguientes parámetros: el número de estimadores (n_estimators), la profundidad máxima del árbol (max_depth), y el número mínimo de muestras por hoja (min_samples_leaf).

Inicialmente, se evaluó el impacto del número de estimadores en el rendimiento del modelo. Se construyeron múltiples modelos de bosques aleatorios, variando el número de estimadores de 10 a 200 en incrementos de 10. Se continuó explorando el efecto de la profundidad máxima del árbol en el rendimiento del modelo. Se ajustaron modelos de bosques aleatorios con diferentes profundidades máximas (2, 10, y 15). Finalmente, se examinó la influencia del número mínimo de muestras por hoja en el desempeño del modelo. Se entrenaron múltiples modelos con diferentes valo-res de este parámetro (1, 3, 5, 10, y 50), manteniendo constante la profundidad máxima.

Los resultados finales muestran que el modelo de bosque aleatorio con 200 estimadores, una profundidad máxima de 10 y un número mínimo de muestras por hoja de 1 logró alcanzar un AUC de 0.9785 en el conjunto de validación. Esta configuración óptima demuestra la importancia de ajustar cuidadosamente los hiperparámetros para obtener un rendimiento óptimo del modelo en la clasificación de muestras de cáncer de próstata.

5.1. Selección de características utilizando BSS/WSS

La selección de características utilizando BSS/WSS permite reducir la dimensionalidad del conjunto de datos y mejorar la eficiencia computacional del mode-lo, al tiempo que conserva la información más relevante para la tarea de clasificación de muestras de cáncer de próstata. Al reducir la redundancia en los datos, esta técnica puede ayudar a mejorar la precisión y la generalización del modelo.

El cálculo de la relación entre la suma de cuadrados entre clases (BSS) y la suma de cuadrados dentro de las clases (WSS) para cada gen en el conjunto de datos permitió evaluar la capacidad de cada gen para discriminar entre clases de muestras. Utilizando los resultados de BSS/WSS, se seleccionaron los 100 mejores genes que contribuyen significativamente a la variabilidad entre las clases. Estos genes seleccionados forman un conjunto de datos reducido que conserva las características más importantes para la clasificación de muestras de cáncer de próstata.

Como anteriormente se realizó con el conjunto de datos completo, se aplicaron los algoritmos de aprendizaje automático, incluidos la regresión logística, árboles de decisión y bosques aleatorios, para realizar la clasificación de muestras de cáncer de próstata únicamente con las características más relevantes. Esta vez el modelo de regresión logística obtuvo una precisión del 91.82 %, una sensibilidad del 94 % y una especificidad del 70 %. Para el árbol de decisión, se exploraron diferentes profundidades máximas y diferentes números mínimos de muestras por hoja. Se calculó el AUC para cada configuración de hiperparámetros, obteniendo los resultados de la Fig. 2.

Se ajustó un modelo de árbol de decisión con una profundidad máxima de 10 y un número mínimo de muestras por hoja de 15 utilizando el conjunto de entrenamiento. Luego, se realizaron predicciones sobre el conjunto de validación y se calculó el AUC, obteniendo un valor de 0.8454.

Para el bosque aleatorio, se observó un aumento en el AUC con el incremento del número de árboles y la profundidad máxima como se puede observar en la Fig. 3. Además, se evaluaron diferentes valores del hiperparámetro de número mínimo de muestras por hoja (Fig. 4) para el bosque aleatorio con una profundidad máxima de 10. Se encontró que la AUC era más alta para el valor igual a 1 de dicho hiperparámetro.

En general, el modelo de bosque aleatorio con 200 estimadores, una profundidad máxima de 10 y un número mínimo de muestras por hoja igual a 1 tuvo el mejor rendimiento con un AUC de 0.925 en el conjunto de validación.

6. Conclusión

La investigación analizó la aplicación de técnicas de selección de características, específicamente basadas en la relación de suma de cuadrados entre grupos y dentro de grupos (BSS/WSS), en el análisis de expresión génica para clasificar muestras de cáncer de próstata. Los resultados revelaron la eficacia de esta técnica para identificar genes relevantes en la clasificación de muestras tumorales y normales.

La selección de características basada en BSS/WSS identificó un conjunto óptimo de genes que contribuyen significativamente a la variabilidad entre las clases de muestras de cáncer de próstata. Esta reducción de la dimensionalidad facilitó el modelado y mejoró la eficiencia computacional de los algoritmos de aprendizaje automático utilizados.

Los modelos de regresión logística, árboles de decisión y bosques aleatorios, luego de la selección de características, mostraron un rendimiento prome-tedor en la clasificación de muestras tumorales y normales, evidenciando la informatividad y relevancia de los genes seleccionados. Además, al ajustar cuidadosamente los hiperparámetros, se observó una mejora en el rendimiento de los modelos de bosques aleatorios. A pesar de reducir el número de genes de más de 20,500 a solo 100, el AUC no disminuyó significativamente.

Esto resalta la importancia de la optimización para lograr un rendimiento óptimo en la clasificación de muestras de cáncer de próstata. Para futuras investigaciones en el campo de la clasificación del cáncer de próstata, se pueden explorar diversas áreas. En primer lugar, sería interesante investigar técnicas avanzadas para seleccionar el número óptimo de características (genes) que realmente contribuyen a la clasificación precisa. Además, se podría comparar el enfoque utilizado en este estudio con otros métodos que emplean cómputo evolutivo, identificando posibles áreas de mejora y validando la efectividad de las características seleccionadas.

También se recomienda probar los modelos y las características en conjuntos de datos externos e independientes, asegurándose de que estén equilibrados para evaluar la generalización del enfoque. Por último, sería valioso crear modelos utilizando los hiperparámetros optimizados en este estudio para verificar su validez en un entorno clínico real, lo que podría tener implicaciones significativas para el diagnóstico y tratamiento del cáncer de próstata.

Agradecimientos. El autor principal agradece el apoyo recibido por el Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) a través del Programa de Becas para Estudios de Posgrado en México.

Referencias

1. Bar, Y., Keenan, J. C., Ryan, L., Juric, D., Shin, J., Wander, S. A., Spring, L. M., Moy, B., Ellisen, L., Isakoff, S. J., Bardia, A., Vidula, N.: Abstract P4-01-14: Changes in the Genomic spectrum of actionable alterations in HER2 negative metastatic breast cancer in serial cell free DNA (cfDNA) analysis. Cancer Research, vol. 83, no. 5_Supplement (2023) doi: 10.1158/1538-7445.SABCS22-P4-01-14
2. Barrué, C.: The i-walker: An intelligent pedestrian mobility aid. Computational Intelligence in Healthcare 4 Studies in Computational Intelligence, vol. 309 (2010) doi: 10.1007/978-3-642-14464-6
3. Bichindaritz, I.: Comparison of reuse strategies for case-based classification in bioinformatics. In: Case-Based Reasoning Research and Development, pp. 393–407 (2011) doi: 10.1007/978-3-642-23291-6_29
4. Deshpande, D., Chhugani, K., Chang, Y., Karlsberg, A., Loeffler, C., Zhang, J., Muszyńska, A., Munteanu, V., Yang, H., Rotman, J., Tao, L., Balliu, B., Tseng, E., Eskin, E., Zhao, F., Mohammadi, P., P. Łabaj, P., Mangul, S.: RNA-seq data science: From raw data to effective interpretation. Frontiers in Genetics, vol. 14 (2023) doi: 10.3389/fgene.2023.997383
5. Farrell, R. E.: Chapter 24 - RNA-seq: The premier transcriptomics tool. RNA Methodologies (Sixth Edition), pp. 697–721 (2023) doi: 10.1016/B978-0-323-90221-2.00045-X
6. Ferlay, J., Colombet, M., Soerjomataram, I., Mathers, C., Parkin, D., Piñeros, M., Znaor, A., Bray, F.: Estimating the global cancer incidence and mortality in 2018: GLOBOCAN sources and methods. International Journal of Cancer, vol. 144, no. 8, pp. 1941–1953 (2019) doi: 10.1002/ijc.31937
7. Loeb, S., Vellekoop, A., Ahmed, H. U., Catto, J., Emberton, M., Nam, R., Rosario, D. J., Scattoni, V., Lotan, Y.: Systematic review of complications of prostate biopsy. European Urology, vol. 64, no. 6, pp. 876–892 (2013) doi: 10.1016/j.eururo.2013.05.049
8. Million, M., Feyissa, T.: RNA-Seq as an effective tool for modern transcriptomics, a review-based study. Journal of Applied Research in Plant Sciences, vol. 3, no. 02, pp. 236–241 (2022) doi: 10.38211/joarps.2022.3.2.29
9. Shen, C. H.: Chapter 3 - Gene expression: Transcription of the genetic code. Diagnostic Molecular Biology (Second Edition), pp. 57–89 (2023) doi: 10.1016/B978-0-323-91788-9.00003-X
10. Subramanian, I., Verma, S., Kumar, S., Jere, A., Anamika, K.: Multi-omics data integration, interpretation, and its application. Bioinformatics and biology insights, vol. 14 (2020) doi: 10.1177/1177932219899051
11. Taylor, B. S., Schultz, N., Hieronymus, H., Gopalan, A., Xiao, Y., Carver, B. S., Arora, V. K., Kaushik, P., Cerami, E., Reva, B., Antipin, Y., Mitsiades, N., Landers, T., Dolgalev, I., Major, J. E., Wilson, M., Socci, N. D., Lash, A. E., Heguy, A., Eastham, J. A., et al.: Integrative Genomic Profiling of Human Prostate Cancer. Cancer Cell, vol. 18, no. 1, pp. 11–22 (2010) doi: 10.1016/j.ccr.2010.05.026
12. Wang, G., Zhao, D., Spring, D. J., DePinho, R. A.: Genetics and biology of prostate cancer. Genes & development, vol. 32, no. 17-18, pp. 1105–1140 (2018) doi: 10.1101/gad.315739.118

ISSN 1870-4069

Ecosistema de internet de las cosas para la clasificación de la calidad del agua mediante aprendizaje máquina

Valentín Calzada-Ledesma, Güily Uziel Cruz-Gallo,
Alan Eduardo Stuart Cabrera-Alcalá, Jonathan López-Arellano

Instituto Tecnológico Superior de Purisima del Rincón,
Ingeniería en Informática,
México

valentin.cl@purisima.tecnm.mx

Resumen. En este artículo, se propone un sistema de monitoreo y clasificación de la calidad del agua utilizando Internet de las Cosas y Aprendizaje Máquina. A través de la instrumentación de un dispositivo gestionando por un microcontrolador ESP32, se obtienen datos de cuatro sensores con los que se caracteriza la calidad del agua, los parámetros medidos son: temperatura, sólidos disueltos totales, turbidez y nivel de pH. Se consolidó un conjunto de datos (que paralelamente se almacena en la nube mediante un protocolo MQTT) con 60,000 registros repartidos equitativamente en seis clases de agua diferentes. Con éste se entrenaron tres modelos de Redes Neuronales Artificiales tipo Perceptrón Multicapa para clasificar diferentes tipos de agua, y a su vez, determinar si su calidad es apta para consumo humano y/o riego, obteniendo (para este diseño experimental propuesto) una exactitud promedio superior al 95 % bajo un esquema de validación cruzada. La implementación de este tipo de tecnologías puede proporcionar información valiosa para facilitar la toma de decisiones en temas relacionados con la gestión y uso del agua.

Palabras clave: Calidad del agua, redes neuronales artificiales, internet de las cosas, reconocimiento de patrones, ESP32.

Internet of Things Ecosystem for Water Quality Classification Using Machine Learning

Abstract. This article proposes a water quality monitoring and classification system using the Internet of Things and Machine Learning. Through the instrumentation of a device managed by an ESP32 microcontroller, data is obtained from four sensors with which the water quality is characterized, the parameters measured are temperature, total dissolved solids, turbidity, and pH level. A data set was consolidated (also stored in the cloud using an MQTT protocol) with 60,000 records evenly distributed in six different water classes. With this data, three Multilayer Perceptron type Artificial Neural Network models were trained to classify different types of water, and in turn, determine whether its quality is suitable for human consumption or irrigation, obtaining (for this proposed experimental design) an average accuracy greater than 95% under a

cross-validation scheme. The implementation of this type of technology can provide valuable information to facilitate decision-making on issues related to water management and use.

Keywords: Water Quality, Artificial Neural Networks, Internet of Things, Pattern Recognition, ESP32.

1. Introducción

El agua es un recurso hídrico indispensable para la existencia de los seres vivos, y aunque es un recurso natural renovable, su uso no se limita únicamente al consumo humano, sino que también se utiliza para actividades de agricultura, ganadería y producción de energía [11]. Es innegable que el agotamiento y deterioro de las fuentes de recursos hídricos es un fenómeno que no puede ser subestimado. La sobreexplotación de acuíferos, el cambio climático y la variabilidad en los patrones de precipitación, entre otros factores, han incrementado la vulnerabilidad de las reservas de agua, llevando a situaciones críticas de escasez y estrés hídrico en distintas regiones del mundo [4]. En este contexto, México no es una excepción.

La medición de la calidad del agua es de vital importancia debido a su impacto directo en la salud humana, la conservación del medio ambiente y la sostenibilidad de los recursos hídricos. En un contexto donde el acceso a agua potable segura es fundamental para la salud y el bienestar de las comunidades, la capacidad de monitorear y clasificar la calidad del agua es crucial para prevenir enfermedades transmitidas por el agua y garantizar un suministro adecuado para consumo humano y actividades agrícolas [5].

Sin embargo, la clasificación de la calidad del agua presenta desafíos significativos, como la detección de contaminantes, la evaluación de la contaminación a lo largo del tiempo y en diferentes fuentes de agua, y la implementación de tecnologías y métodos de monitoreo adecuados para abordar la complejidad de los ecosistemas acuáticos. Sin embargo, este tipo de análisis usualmente se realizan en laboratorios especializados con equipo profesional, a los cuales puede ser difícil acceder en zonas remotas.

Por otro lado, la falta de datos actualizados y la infraestructura de monitoreo limitada son también obstáculos importantes que dificultan la toma de decisiones informadas y la implementación efectiva de medidas de protección y conservación del agua [1]. En este sentido, es imperativo desarrollar estrategias de monitoreo y clasificación de la calidad del agua, para abordar estos desafíos y asegurar la disponibilidad de agua limpia y saludable para las generaciones presentes y futuras.

Para lidiar con el problema de la falta de laboratorios especializados para el análisis de la calidad del agua, en [3] se estable que sensores de bajo costo integrados con Internet de la Cosas (IoT por sus siglas en inglés), pueden utilizarse para realizar un monitoreo ambiental del agua, dejando claro que al ser dispositivos de bajo costo es necesaria más investigación sobre su precisión y confiabilidad en comparación con equipos profesionales. Por otro lado, en [9] se discute el cómo los ecosistemas de IoT en el contexto del monitoreo de la calidad del agua, pueden aumentar la conciencia social al proporcionar datos accesibles en tiempo real, lo cual es positivo en zonas con

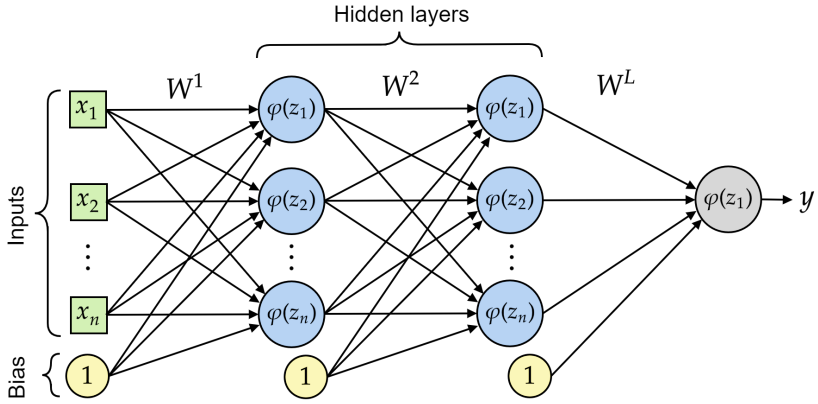


Fig. 1. Red Neuronal Artificial tipo Perceptrón Multicapa.

estrés hídrico. Además, en [12] se propone el uso de Redes Neuronales Artificiales (RNA) para detectar la calidad del agua logrando resultados satisfactorios. Asimismo, en [10] se propuso el modelo CATBoost, ofreciendo una buena precisión para tareas de clasificación de datos de calidad del agua, lo que sugiere que los algoritmos de aprendizaje máquina pueden ser un enfoque confiable para mejorar el monitoreo de la calidad del agua mediante inteligencia artificial.

Motivados por los trabajos antes mencionados, y la necesidad de implementar sistemas relacionados con la cuarta revolución industrial, en este artículo se propone un sistema de Internet de las Cosas (IoT por sus siglas en inglés) para monitorear y clasificar la calidad del agua en tiempo real, esto a través de un sistema de clasificación dirigido por una RNA tipo Perceptrón Multicapa (MLP por sus siglas en inglés). La elección de utilizar un MLP en nuestro estudio se respalda por un análisis en el estado del arte, en donde este tipo de RNAs son ampliamente utilizadas para lidiar con problemas relacionados con la calidad del agua [7, **?**,**?**].

Además, para nuestro caso de estudio, este tipo de RNA presenta versatilidad y una alta capacidad para modelar relaciones no lineales entre variables de entrada y salida, es fácil de implementar, y los modelos de creados tienden a ser más ligeros en términos de tamaño y complejidad computacional. Este último factor es fundamental para nuestra implementación, ya que en comparación con algunas otras arquitecturas de RNAs, como las Redes Neuronales Convolucionales (CNN) o las Redes Neuronales Recurrentes (RNN), un MLP suele ser más rápido en su fase de inferencia. Lo cual es crucial para una aplicación en tiempo real de IoT con conexión a la nube.

La presente propuesta podría representar un avance tecnológico significativo hacia un futuro del uso del agua más eficiente, sostenible y benéfico para la sociedad en general. El artículo se organiza de la siguiente manera. En la Sección 2, se presentan los conceptos relacionados con el trabajo. En la Sección 3, se presenta la metodología. El diseño de experimentos se muestra en la Sección 4. Los resultados se reportan y analizan en la Sección 5. En la Sección 6, se exponen las limitaciones del trabajo seguidas del trabajo futuro. Finalmente, se muestran las conclusiones en la Sección 8.

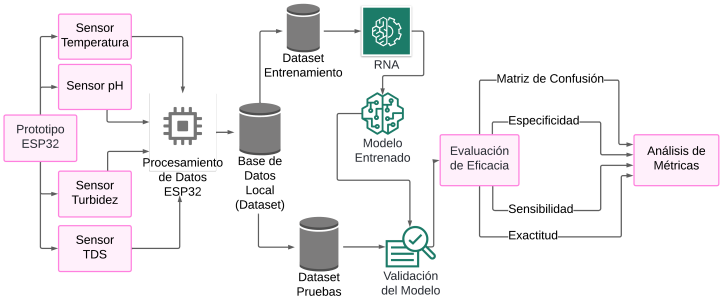


Fig. 2. Esquema general del proceso de aprendizaje máquina.

2. Marco teórico

2.1. Recursos hídricos

Los recursos hídricos cumplen un papel muy importante en nuestra vida diaria, al proporcionar agua dulce para satisfacer las diferentes actividades humanas, sustentar la biodiversidad, y además, para equilibrar el ecosistema [15]. Entre las actividades esenciales humanas se encuentran la agricultura, el consumo humano, la generación de energía, la ingeniería ganadera, y finalmente, la industria química.

Sin embargo, los recursos hídricos cada vez escasean y son vulnerables frente a la contaminación y el cambio climático, por lo que su gestión sostenible es uno de los principales retos que se abordan en la actualidad en México, para garantizar la equidad en el acceso al agua, la protección de los ecosistemas acuáticos y la resiliencia ante los desafíos futuros, como el estrés hídrico.

2.2. Redes neuronales artificiales

Una Red Neuronal Artificial (RNA) es una abstracción matemática inspirada en el funcionamiento del sistema nervioso biológico, específicamente en la forma en que las neuronas interactúan y procesan información [6]. El objetivo principal de una RNA es "aprender" a partir de datos de ejemplo, para realizar tareas específicas como el reconocimiento de patrones o la predicción.

Un tipo de RNA es el Perceptrón Multicapa (MLP por sus siglas en inglés), el cual está compuesto por un conjunto de "neuronas" organizadas en capas interconectadas llamadas capas ocultas (*hidden layers*). Desde un punto de vista matemático, un MLP puede definirse como un grafo dirigido que consta de un conjunto de nodos y de conexiones entre los mismos [6]. La operación matemática básica que realiza un MLP es la siguiente:

$$y = \varphi(Z), \tag{1}$$

donde y es la salida de la neurona, $\varphi(\cdot)$ es una función de activación, usualmente la función ReLU (*Rectified Linear Units*), y Z es un vector (columna) de pre-activaciones z_i que se calcula de la siguiente manera:

$$Z = W \cdot X + b. \tag{2}$$

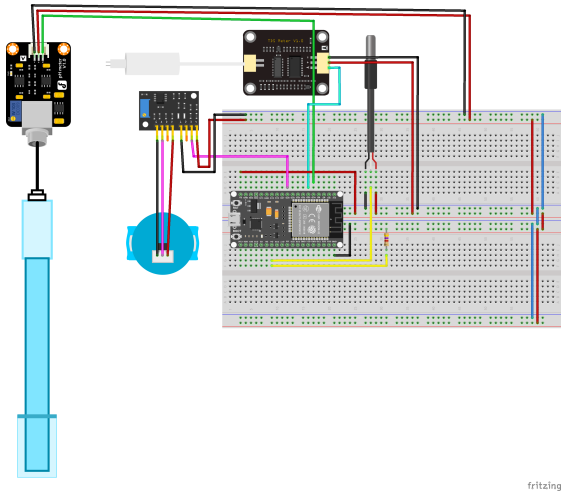


Fig. 3. Instrumentación del ecosistema IoT para el sensado de parámetros.

En este contexto, b es el sesgo (usualmente un vector de 1's), X es un vector columna de n entradas x_i, y W^l es una matriz cuadrada de pesos para cada capa $l \in [1, L]$. En donde cada fila m de la matriz, corresponde a un vector de n pesos de un perceptrón simple o neurona. En la Figura. 1 se muestra el esquema general de un MLP. El algoritmo básico para entrenar a un MLP se llama retropropagación, el cual es un método de optimización que permite ajustar los pesos W de las conexiones entre las neuronas de la RNA para minimizar una función de pérdida, lo que en última instancia mejora el rendimiento del modelo en la tarea que está realizando. Para más información, consulte [13].

2.3. Sensores para medir la calidad del agua

En México, existe la norma mexicana NOM-127-SSA1-1994: "Agua para uso y consumo humano, límites permisibles de calidad y tratamientos a que debe someterse el agua para su potabilización", y la NOM-CCA/032-ECOL/1993: "Límites máximos permisibles de contaminantes en las aguas residuales de origen urbano o municipal para su disposición mediante riego agrícola", en ambas normas se establecen los parámetros fundamentales para considerar si el agua es apta o no para consumo humano y para riego. De acuerdo con los parámetros establecidos en las normas, y a su vez, realizando un contraste de los sensores disponibles en el mercado, los parámetros elegidos para evaluar la calidad del agua en este trabajo son: temperatura, sólidos disueltos totales, turbidez y nivel de pH. Estos cuatro parámetros proporcionan una evaluación básica pero importante de la calidad del agua y pueden indicar la presencia de contaminantes, cambios ambientales o condiciones que afectan su idoneidad para el consumo humano. Además, son parámetros que pueden medirse de manera relativamente sencilla y económica, lo que los hace adecuados para monitoreo continuo en tiempo real o en sistemas de detección temprana de problemas de calidad del agua. A continuación, se describen cada uno de éstos.

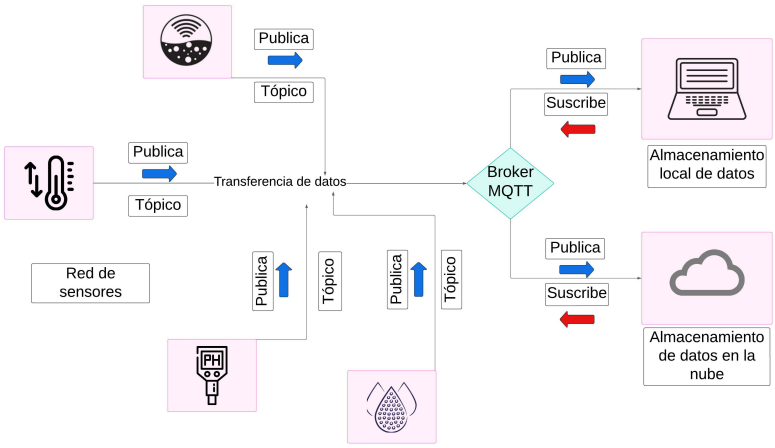


Fig. 4. Diagrama de comunicación MQTT.

– **Sensor de temperatura DS18B20:** Este sensor es capaz de realizar mediciones de la temperatura del agua [14] en un amplio rango de temperaturas desde los -55°C hasta los 125°C, con una alta precisión en sus mediciones. Una de sus características principales es que es sumergible.

– **Sensor de sólidos disueltos totales:** Este sensor indica cuantos miligramos de sólidos solubles se disuelven en un litro de agua. Esto se hace a través del parámetro TDS (Total de Sólidos Disueltos), en el cual, mientras mayor sea su valor, indicará que más sólidos solubles se encuentran dentro del agua, lo que sugiere que el agua está menos limpia [2].

– **Sensor de pH:** Este instrumento ayuda a medir la actividad de iones-hidrógeno en las soluciones basadas en agua, esto se hace a través del parámetro pH que indica la acidez o alcalinidad del agua. Para lograrlo, este sensor mide los niveles de pH usando el potencial eléctrico entre un electrodo de pH y un electrodo de referencia.

– **Sensor de turbidez TSW-20M:** Es un sensor analógico tipo óptico capaz de medir los niveles de turbieza u opacidad que hay en los líquidos, particularmente agua [2]. Para lograr esto, el sensor detecta las partículas suspendidas en el agua, a través de la tasa de dispersión de la luz.

2.4. Internet de las cosas

En el contexto de la gestión de la calidad del agua, la integración de la tecnología de Internet de las Cosas (IoT por sus siglas en inglés) ofrece una solución innovadora para la recopilación de datos en tiempo real utilizando sensores. En esta sección se exploran tres componentes clave del ecosistema de IoT que juegan un papel fundamental para medir la calidad del agua: el protocolo de comunicación MQ Telemetry Transport (MQTT), la base de datos InfluxDB y la plataforma de integración Node-RED.

Estas herramientas trabajan en conjunto para permitir la captura y el almacenamiento de datos precisos sobre la calidad del agua. Al unir estas tecnologías, se facilita la monitorización continua y la toma de decisiones informadas para la preservación y gestión efectiva de los recursos hídricos.

El Protocolo de Comunicación MQTT, se destaca como el principal protocolo de mensajería para el Internet de las Cosas (IoT), facilitando la publicación y suscripción de datos entre dispositivos IoT a través de la red. Operando bajo el patrón de publicación/suscripción (Pub/Sub), MQTT conecta emisores y receptores mediante tópicos, permitiendo su desacoplamiento y gestionando la conexión a través de un intermediario conocido como broker MQTT. Por otro lado, InfluxDB, una base de datos desarrollada por InfluxData en Go, garantiza un almacenamiento eficiente y una rápida recuperación de datos, especialmente de series temporales, ampliando su aplicación desde la monitorización hasta el IoT y el análisis de datos de sensores. Finalmente, Node-RED emerge como una herramienta de programación innovadora, simplificando la conexión entre dispositivos de hardware, APIs y servicios en línea.

3. Metodología

En esta sección, se describe la metodología utilizada en este trabajo de investigación. De manera general, se instrumenta un dispositivo IoT que sensa cuatro parámetros sobre la calidad de seis tipos de agua diferente. Después, se consolida un conjunto de datos para entrenar un MLP, posteriormente se valida el modelo, y finalmente se obtienen diferentes métricas para conocer su desempeño. En la Figura 2 se muestra el esquema general de la metodología.

3.1. Obtención de datos

Los datos se obtuvieron a través de un dispositivo orquestado por un microcontrolador ESP32 instrumentado con cuatro sensores: el DS18B20 para medir la temperatura del agua, un sensor TDS para medir los sólidos disueltos totales, un sensor de turbidez TSW-20M, y un sensor de pH. Estos sensores se utilizaron para medir y obtener una caracterización cuantitativa de seis tipos de agua diferentes: agua limpia, agua con detergente, agua del grifo, agua con pesticida, agua con cloro y agua de pecera. En la Figura 3, se muestra la instrumentación realizada del dispositivo IoT.

Las mediciones se realizaron bajo un entorno controlado, se utilizaron recipientes de plástico para almacenar las muestras de cada tipo de agua a temperatura ambiente. Para el registro de las lecturas de los sensores se utilizó Arduino IDE, recopilando un total de 10,000 registros por clase de agua, los cuales se almacenaron en un archivo .CSV para su posterior procesamiento en el lenguaje de programación Python, asegurando así una amplia variedad de datos para el posterior entrenamiento del MLP.

Paralelamente, los datos sensados se transmiten mediante un protocolo de comunicación M2M (Machine-to-Machine) MQTT, y son almacenados en la nube utilizando un sistema de gestión de bases de datos de series temporales llamado InfluxDB.

Tabla 1. Muestras de datos obtenidos por los sensores, reescalados entre 0 y 1.

Temperatura	TDS	Turbidez	PH	Clase	Consumo Humano	Riego
0.2104	0.1948	0.0153	0.3027	A	Sí	Sí
0.3687	0.9772	0.7332	0.7891	B	No	No
0.0421	0.1448	0.0328	0.6141	C	No	Sí
0.9899	0.5325	0.3911	0.8006	D	No	No
0.9360	0.1910	0.2616	0.0072	E	No	No
0.8215	0.3144	0.4183	0.2296	F	No	Sí

Este sistema se eligió debido a que puede funcionar en sistemas distribuidos, puede ser accesible a través de protocolos estándar de red, y finalmente, su nivel de seguridad permite autenticación de usuarios. La Figura 4 muestra el diagrama completo de este sistema IoT. Cabe señalarse que hasta este punto del proyecto sólo se realizó el almacenamiento de las series de tiempo, y posteriormente, con esa información se propondrán aplicaciones basadas en la nube, sin embargo, esto se enmarca como trabajo futuro.

4. Diseño de experimentos

4.1. Conjunto de datos

El conjunto de datos consta de 60,000 instancias, cuatro características (una por sensor), y seis clases diferentes de agua. Cada instancia está etiquetada con una clase (designada de la A a la F). Asimismo, se añaden dos variables objetivo (bi-clase) adicionales: “Consumo humano” y “Riego”, las cuales se basan en las normas NOM-127-SSA1-1994 y NOM-CCA/032-ECOL/1993 respectivamente (descritas en la sección Sección 2.3), ampliando la utilidad del conjunto de datos al permitir la evaluación de la calidad del agua para diferentes usos. Esto es de suma utilidad, ya que a partir del mismo conjunto de datos, se pueden entrenar diferentes modelos, uno para cada una de las variables objetivo. Este procedimiento se detalla más adelante.

Es importante mencionar que los datos obtenidos fueron reescalados entre 0 y 1 utilizando el método MinMaxScaler de la librería scikit-learn. Este reescalamiento es importante, ya que asegura que los vectores de características se encuentren en el mismo dominio, permitiendo que el proceso de entrenamiento del MLP sea adecuado. En el Tabla 1 se exponen 6 muestras de datos normalizados, su clase correspondiente, si es apto para consumo humano y si es apto para riego.

4.2. Parámetros de los algoritmos

Como se mencionó anteriormente, el conjunto de datos consta de tres variables objetivo: `Clase_Agua`, `Consumo_Humano` y `Riego`, por lo que se entrenó un modelo de RNA independiente para cada una de estas variables. Además, con la finalidad de conocer la robustez de las arquitecturas de las RNA propuestas, se utilizó

Tabla 2. Arquitecturas de RNA para cada variable objetivo.

Variable objetivo	Arquitectura RNA
Clase_Agua	4, 5, 5, 6
Consumo_Humano	4, 8, 7, 6
Riego	4, 5, 5, 6

un esquema de validación cruzada con $K = 10$ folds (pliegues) [8]. A continuación, se muestra la configuración de cada una de las arquitecturas. El entrenamiento de los modelos se realizó utilizando el algoritmo de retropropagación y la función de activación ReLU (Rectified Linear Unit). Se estableció un máximo de 1000 épocas para ajustar los pesos de la RNA y minimizar la función de pérdida, en este caso, el error cuadrático medio (MSE por sus siglas en inglés).

Como se mencionó anteriormente, la evaluación del rendimiento de cada modelo se realizó mediante validación cruzada, por lo que se reportan las gráficas del comportamiento de la función de pérdida durante el proceso de entrenamiento, los resultados de exactitud para cada Fold, y la exactitud media de la validación cruzada, para cada variable objetivo. Finalmente, el modelo entrenado se guardó en un archivo utilizando la función joblib.dump de scikit-learn para su posterior uso en un contexto de IoT en tiempo real.

5. Resultados

Durante el proceso de entrenamiento y evaluación de los modelos de RNA, se obtuvieron resultados para las tres variables objetivo: `Clase_Agua`, `Consumo_Humano` y `Riego`, los cuales se reportan en el Tabla 3. Asimismo, en la Figura 5, se muestran las gráficas del comportamiento de la función de pérdida durante el proceso de entrenamiento de cada uno de los modelos. En la Figura 5, se puede observar cómo las funciones de pérdida convergen en pocas iteraciones, sugiriendo eficacia en los modelos entrenados bajo este diseño experimental.

La alta exactitud, sensibilidad y especificidad, exhibidas en la Tabla 3, indican que los modelos son capaces de capturar los patrones distintivos en los datos de los sensores y utilizarlos para realizar la clasificación de los tipos de agua, y determinar si son áptas para consumo humano y riego. Este nivel de rendimiento es prometedor y sugiere que los modelos pueden ser herramientas útiles para monitorear y evaluar la calidad del agua en diversos entornos.

Es importante mencionar que al utilizar validación cruzada, se puede obtener una estimación más precisa del rendimiento de los modelos, ayudando a evitar su sobreajuste, ya que se exponen a diferentes subconjuntos de datos durante el entrenamiento y la validación.

Con respecto a las arquitecturas propuestas, dichas configuraciones se obtuvieron después de un proceso de experimentación exhaustiva, en donde se observó que con el hecho de reducir la cantidad de neuronas en alguna de las capas ocultas, significaba un decremento de la exactitud promedio de hasta un 30 %. Por otro lado, si se incrementaba la cantidad de neuronas o la cantidad de capas ocultas, el rendimiento

Tabla 3. Resultados de la validación cruzada para las variables objetivo `Clase_Agua`, `Consumo_Humano`, y `Riego`.

Fold	Clase_Agua	Consumo_Humano	Riego
1	1.00	0.99	1.00
2	1.00	0.99	1.00
3	1.00	0.99	1.00
4	1.00	0.99	1.00
5	1.00	0.99	1.00
6	1.00	0.99	1.00
7	1.00	0.99	1.00
8	1.00	0.99	1.00
9	1.00	0.99	1.00
10	1.00	0.99	1.00
Exactitud media	**100.00 %**	**99.07 %**	**99.90 %**
Sensibilidad media	**100.00 %**	**98.95 %**	**99.90 %**
Especificidad media	**100.00 %**	**98.95 %**	**99.90 %**

obtenido no era significativamente mejor, aunque sí incrementaba significativamente el costo computacional. Por lo que, una arquitectura simple, para el contexto de nuestro trabajo, es mejor, ya que se adapta a las necesidades de un sistema de IoT en tiempo real.

Finalmente, se conjetura que los datos obtenidos a través de esta metodología, podrían ser linealmente separables, este hecho es algo que no se esperaba al realizar este proyecto, puesto que intuitivamente se esperaría que los datos estuvieran correlacionados. Sin embargo, explicaría el buen desempeño de la RNA, esto es algo que estudiaremos a detalle en un trabajo futuro.

6. Limitaciones

Durante el desarrollo del proyecto, se identificaron diversas limitaciones que podrían afectar el diseño y la implementación del sistema de monitoreo de la calidad del agua:

– **Volumen de datos limitado:** Para este artículo se obtuvieron un total de 60,000 registros para consolidar el conjunto de datos. Sin embargo, conjeturamos que a medida de que crezca el número de clases, probablemente habrá una disminución en el desempeño del modelo, por lo que será necesario entrenar otro modelo.

– **Inexactitud en las mediciones de los sensores:** La medición de los sensores podría ser imprecisa debido a factores como la temperatura del agua, la intensidad de la luz sobre el agua y los voltajes recibidos por los sensores, estos aspectos pueden influir en los resultados de las mediciones. Además, cambios bruscos entre diferentes tipos de agua pueden afectar la estabilidad y precisión de las lecturas de los sensores, ya que algunos de pestos pueden requerir más tiempo para adaptarse o estabilizarse.

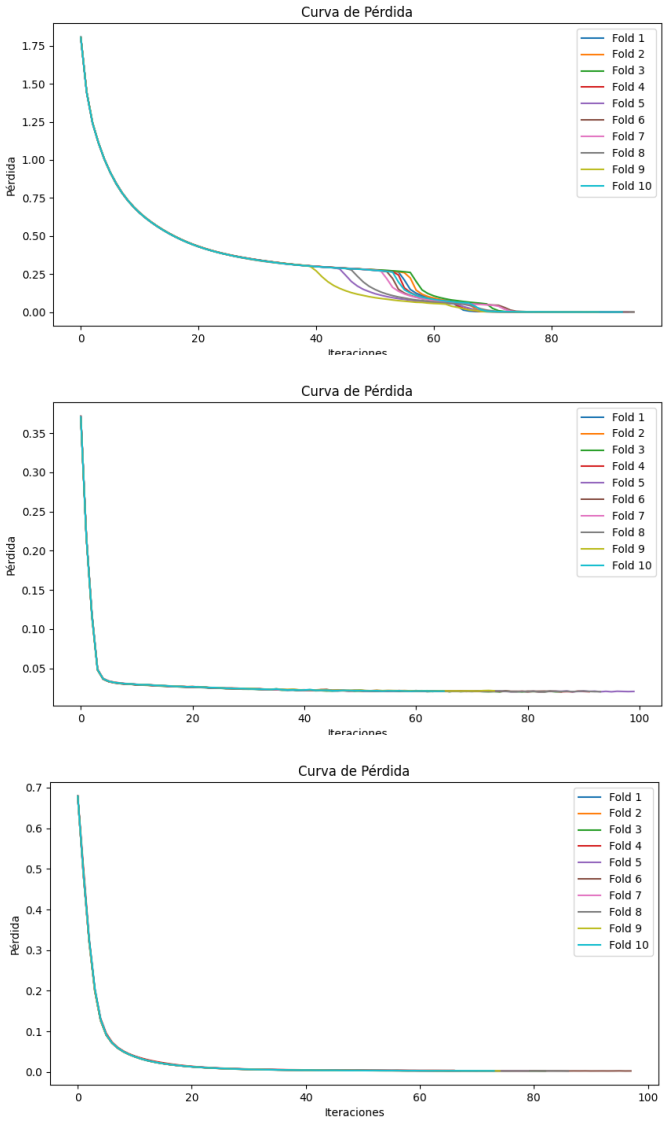


Fig. 5. Comportamiento de la función de pérdida para `Clase_Agua`, `Consumo_Humano` y `Riego`, respectivamente.

7. Trabajo futuro: hacia el cómputo distribuido MPI

En la actualidad, existe una gran variedad de algoritmos que requieren de muchos recursos computacionales para llevar a cabo sus tareas debido a la gran cantidad de procesamiento de datos que se manipulan. Estos tipos de algoritmos necesitan grandes capacidades de procesamiento de datos, que una computadora convencional

e incluso las que están específicamente diseñadas para abordar esas tareas no son suficientes para satisfacer las necesidades o explotar el potencial de las aplicaciones. Para llevar a cabo el entrenamiento de una RNA bajo un esquema de cómputo distribuido utilizando MPI, será necesario recopilar y preparar los datos de todos los sensores, organizándolos y adecuándolos para su procesamiento en un clúster de cómputo distribuido. Posteriormente, se implementarán técnicas de cómputo distribuido utilizando bibliotecas y herramientas especializadas, como TensorFlow o MPI. Estas herramientas permitirán distribuir la carga de trabajo entre los nodos del clúster, lo que acelerará significativamente el proceso de entrenamiento de la Red Neuronal.

8. Conclusiones

La medición precisa de la calidad del agua es crucial para garantizar la salud humana, la conservación del medio ambiente y la sostenibilidad de los recursos hídricos. A pesar de su importancia, enfrenta desafíos significativos, como la detección de contaminantes y la falta de datos actualizados y una infraestructura de monitoreo adecuada. El sistema propuesto en el presente artículo, basado en IoT y aprendizaje máquina, ofrece una solución innovadora para el monitoreo de la calidad del agua en tiempo real.

Los resultados del modelo de la RNA, bajo un esquema de validación cruzada, mostraron una exactitud, sensibilidad y especificidad superiores al 95 %, lo cual indica una alta eficacia para clasificar muestras de agua bajo el esquema experimental propuesto. La arquitectura de la RNA es de bajo costo computacional, por lo que puede implementarse sin dificultado en ecosistema de IoT en tiempo real con transferencia de datos a la nube. La implementación de este tipo de tecnologías podría beneficiar a la sociedad en general, al garantizar la disponibilidad de alimentos de alta calidad y promover la seguridad alimentaria, y a su vez, podría contribuir a un futuro de la gestión del agua más eficiente y sostenible.

Agradecimientos. Se agradece al Tecnológico Nacional de México por el financiamiento brindado que hizo posible la realización de este trabajo. Además, se extiende el reconocimiento al ITS de Purísima del Rincón por su invaluable apoyo, fundamental para culminar con éxito este proyecto.

Referencias

1. Balasooriya, B. K., Rajapakse, J., Gallage, C.: A review of drinking water quality issues in remote and indigenous communities in rich nations with special emphasis on australia. Science of The Total Environment, vol. 903, pp. 166559 (2023) doi: 10.1016/j.scitotenv.2023.166559
2. Conejeros-Molina, A., Hueichaqueo-Pichunman, C., Martinez-Jimenez, B. L., PlaceresRemior, A.: Monitoreo de calidad del agua en sistema de agua potable rural. Ingeniería Electrónica, Automática y Comunicaciones, vol. 42, no. 3, pp. 60–70 (2021)
3. de-Camargo, E. T., Spanhol, F. A., Slongo, J. S., da Silva, M. V. R., Pazinato, J., de-Lima-Lobo, A. V., Coutinho, F. R., Pfrimer, F. W. D., Lindino, C. A., Oyamada, M. S.: Low-cost water quality sensors for IoT: A systematic review. Sensors, vol. 23, no. 9, pp. 4424 (2023) doi: 10.3390/s23094424

4. Du-Plessis, A.: Current and future water scarcity and stress. Water as an inescapable risk: current global water availability, quality and risks with a specific focus on South Africa, pp. 13–25 (2019) doi: 10.1007/978-3-030-03186-2_2
5. Forget, G., Sanchez-Bain, W. A.: Managing the ecosystem to improve human health: Integrated approaches to safe drinking water. International Journal of Occupational and Environmental Health, vol. 5, no. 1, pp. 38–50 (1999) doi: 10.1179/oeh.1999.5.1.38
6. Gardner, M. W., Dorling, S. R.: Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. Atmospheric environment, vol. 32, no. 14-15, pp. 2627–2636 (1998) doi: 10.1016/S1352-2310(97)00447-0
7. Juna, A., Umer, M., Sadiq, S., Karamti, H., Eshmawi, A. A., Mohamed, A., Ashraf, I.: Water quality prediction using KNN imputer and multilayer perceptron. Water, vol. 14, no. 17, pp. 2592 (2022) doi: 10.3390/w14172592 https://www.mdpi.com/2073-4441/14/17/2592
8. Kärkkäinen, T.: On cross-validation for MLP model evaluation. In: Structural, Syntactic, and Statistical Pattern Recognition, pp. 291–300 (2014) doi: 10.1007/978-3-662-44415-3_30
9. Miller, M., Kisiel, A., Cembrowska-Lech, D., Durlik, I., Miller, T.: IoT in water quality monitoring—are we really here? Sensors, vol. 23, no. 2, pp. 960 (2023) doi: 10.3390/s23020960
10. Nasir, N., Kansal, A., Alshaltone, O., Barneih, F., Sameer, M., Shanableh, A., Al-Shamma'a, A.: Water quality classification using machine learning algorithms. Journal of Water Process Engineering, vol. 48, pp. 102920 (2022) doi: 10.1016/j.jwpe.2022.102920
11. Owusu, P. A., Asumadu-Sarkodie, S., Ameyo, P.: A review of ghana's water resource management and the future prospect. Cogent Engineering, vol. 3, no. 1, pp. 1164275 (2016)
12. Rodriguez-Perez, J., Leigh, C., Liquet, B., Kermorvant, C., Peterson, E., Sous, D., Mengersen, K.: Detecting technical anomalies in high-frequency water-quality data using artificial neural networks. Environmental Science & Technology, vol. 54, no. 21, pp. 13719–13730 (2020) doi: 10.1021/acs.est.0c04069
13. Rojas, R.: The backpropagation algorithm. Neural Networks: A Systematic Introduction, pp. 149–182 (1996) doi: 10.1007/978-3-642-61068-4_7
14. Sierra García, L. A.: Diseño del sistema de medición y despliegue de temperaturas con el sensor DS18B20 mediante el protocolo de transmisión 1-WIRE. Ph.D. thesis, Universidad de San Carlos de Guatemala (2017)
15. Torregrosa, M. L., Mora, R. D., Cisneros, B. J., Michel, E. K., Austria, P. M., Cedillo, J. L. M., Viqueira, J. P., Calleros, A. R., Monjardín, L. C. R., Martelo, E. Z.: Los recursos hídricos en México. Diagnóstico del Agua en las Américas, pp. 309 (2012)

Electronic edition
Available online: http://www.rcs.cic.ipn.mx

http://rcs.cic.ipn.mx

Centro de Investigación
en Computación