

Extensión del algoritmo de codificación-decodificación basado en PWM para Redes Neuronales de Impulsos.

Lucas, S.^{a,*}, Portillo, E.^a, Guérin, L.^b, Cabanes, I.^a

^aDepartamento de Ingeniería de Sistemas y Automática, Escuela de Ingeniería de Bilbao, Universidad del País Vasco (UPV/EHU), C/ Plaza Ingeniero Torres Quevedo n° 1, 48013, Bilbao, España.

^bDepartamento de Informática, ENSEIRB-MATMECA, C/ 1 Avenue du Dr Albert Schweitzer, 33400, Talence, Francia.

To cite this article: Lucas, S., Portillo, E., Guérin, L., Cabanes, I. 2023. Extension of the PWM-based encoding-decoding algorithm for Spiking Neural Networks. XLIV Jornadas de Automática, 168-173.
<https://doi.org/10.17979/spudc.9788497498609.168>

Resumen

Las Redes Neuronales de Impulsos (*Spiking Neural Networks*, SNN) son la última generación de redes neuronales y tratan de imitar con mayor fidelidad el funcionamiento del cerebro humano codificando la información a través de *spikes* o series de impulsos. Debido a que la mayoría de procesos reales son analógicos, para emplear este tipo de redes es necesario el uso de algoritmos de codificación y decodificación. El algoritmo de codificación basado en PWM es un novedoso algoritmo temporal de codificación que supera con creces a sus algoritmos predecesores en la precisión a la hora de construir y reconstruir la señal original. A pesar de sus múltiples ventajas, este algoritmo necesita dos puntos cronológicos de la serie temporal original para poder codificar. En este sentido, resulta de interés poder aplicar este tipo de codificación en otro tipo de aplicaciones, como el tratamiento de imágenes, en las que no existe orden cronológico. Por tanto, en este trabajo se presenta una extensión de este algoritmo de codificación para que no sea necesario el uso de dos valores temporales consecutivos y así poder aplicarlo a cualquier tipo de aplicación. Además, la nueva extensión permite reducir en más de un 50 % el coste computacional de los procesos de codificación y decodificación.

Palabras clave: Diseño de metodologías, Validación de modelos, Redes Neuronales, Formulación de modelos, diseño de experimentos, Modelado de series temporales.

Extension of the PWM-based encoding-decoding algorithm for Spiking Neural Networks.

Abstract

Spiking Neural Networks (SNN) are the latest generation of neural networks and attempt to mimic human brain functioning more closely by encoding the information through spike trains. Since most of the real processes are analog, SNN requires the use of encoding-decoding algorithms. The PWM-based encoding-decoding algorithm is a novel temporal encoding algorithm that surpasses its predecessor algorithms in terms of precision. Despite its many advantages, this algorithm requires two chronological values from the original time series in order to encode a spike. In this sense, it is also interesting to be able to apply this algorithm to other types of application, such as image processing, where it does not exist a chronological order of the points. Hence, this paper proposes an extension of the PWM-based encoding-decoding algorithm, in which is not necessary to employ two consecutive values in the encoding process, enabling the algorithm to be applied to any type of application. In addition, the new extension reduces the computational cost of encoding and decoding processes by more than 50 %.

Keywords: Design methodologies, Model validation, Neural Networks, Model formulation, experiment design, Time series modelling.

*Autor para correspondencia: sergio.lucas@ehu.eus
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

1. Introducción

Las Redes Neuronales de Impulsos (*Spiking Neural Networks*, SNN) o también conocidas como la tercera generación de redes neuronales artificiales son modelos neuronales que tratan de imitar con mayor fidelidad el comportamiento de las neuronas biológicas. Para ello, a diferencia de las Redes Neuronales Artificiales tradicionales, las SNN codifican la información mediante series de impulsos o *spikes*. Este tipo de codificación implica que la información no va contenida en la forma de la acción potencial, sino en el número y la distribución temporal de los impulsos Gerstner and Kistler (2002).

La mayoría de los procesos reales son analógicos, por tanto, para poder aplicar las SNN a problemas reales es necesario recurrir a técnicas o algoritmos de codificación, los cuales se encargan de transformar los datos analógicos de los procesos reales en *spikes*. Además, en múltiples aplicaciones de ámbitos como el industrial o la bioingeniería, resulta necesario poder obtener la salida de la red en las unidades/magnitudes correspondientes. Por tanto, las técnicas o algoritmos de codificación también es importante que ofrezcan la posibilidad de decodificar con precisión la salida de las SNN.

Actualmente, dentro de los algoritmos de codificación existentes se pueden diferenciar dos enfoques: codificación basada en frecuencia (*rate coding*) o en tiempo (*temporal coding*). La codificación frecuencial se basa en codificar la información basándose en el número de *spikes* dentro de una ventana de tiempo. Dentro de este enfoque, algunos de los algoritmos más utilizados son: la distribución de Poisson Xu et al. (2017), *Ben's Spike Algorithm* (BSA) Brusca et al. (2019) o *Adaptive Threshold-Based* (ATB) Laña et al. (2018). Por otro lado, la codificación temporal emplea la distancia que hay entre *spikes* para codificar la información. En este caso, destacan los algoritmos de *Time-to-First-Spike* Chen et al. (2016) o *Rank Code* Hong et al. (2020).

Pese a que estudios recientes han afirmado que la codificación temporal preserva mayor cantidad de información que la codificación frecuencial Lopes-dos Santos et al. (2015), esta clase de técnicas a menudo no contempla el proceso de decodificación o el método de decodificación da lugar a errores significativos. En este sentido, en Arriandiaga et al. (2020) y Lucas et al. (2021) se presentó un novedoso algoritmo de codificación-decodificación temporal basado en la modulación por ancho de pulso (*PulseWidth Modulation*, PWM) que supera con creces la precisión en la codificación y decodificación de sus algoritmos predecesores. A pesar de sus múltiples ventajas, este algoritmo requiere de dos puntos cronológicamente consecutivos de la serie temporal para codificar un sólo *spike*. Además, el requerimiento de puntos cronológicamente ordenados hace que este algoritmo no se pueda aplicar directamente a otro tipo de aplicaciones, como el tratamiento de imágenes. En este sentido, en este trabajo se presenta una extensión del algoritmo de codificación-decodificación basado en PWM con el objetivo de solventar estas limitaciones.

El resto del artículo está estructurado de la siguiente forma: en la sección 2, se presenta la propuesta de extensión del algoritmo de codificación-decodificación basado en PWM. En la sección 3, se valida la nueva versión del algoritmo (a) comparando su funcionamiento con la versión original en dos series

temporales diferentes y (b) comprobando su correcto funcionamiento en el procesamiento de imágenes. En la sección 4 se realiza la discusión a la vista de los resultados obtenidos. Por último, en la sección 5 se exponen las conclusiones del trabajo.

2. Extensión algoritmo de codificación basado en PWM.

2.1. Limitaciones en el diseño original del algoritmo PWM.

La idea subyacente en la forma de codificar y decodificar del algoritmo presentado en Arriandiaga et al. (2020) Lucas et al. (2021) reside en los fundamentos teóricos de la metodología PWM. Este algoritmo se basa en emplear una única señal portadora o *carrier* como base temporal para localizar cuándo se dan las intersecciones entre la propia señal y a) la serie temporal original, lo cual da lugar a los *spikes* de entrada a la SNN (fase de codificación); y b) una secuencia de *spikes*, lo cual resulta en la estimación de puntos discretos de la señal original reconstruida (fase de decodificación).

En este sentido, definir correctamente la señal portadora es esencial para la metodología y para ello se emplean dos hiperparámetros: a) el número de ondas de la señal portadora (*number of carrier waves*, nc), que define el número de intersecciones que se podrán dar entre las diferentes señales; y b) el número de puntos de onda de la señal portadora (*number of points per carrier wave*, npc), cuyo valor define la resolución de la codificación y decodificación.

El algoritmo de codificación-decodificación basado en PWM presenta importantes ventajas respecto a otros algoritmos temporales en cuanto a la simplicidad del método, la precisión obtenida en la reconstrucción de señales y la oportunidad de fijar por el usuario mediante el hiperparámetro npc un compromiso entre el coste computación del método y su resolución. Sin embargo, este algoritmo también presenta una serie de limitaciones, especialmente en la parte de la codificación. Tal y como se puede observar en la Figure 1, el algoritmo en su versión original para codificar requiere de la interpolación de dos puntos consecutivos de la serie temporal para poder calcular el instante de intersección con la señal portadora. Este hecho: a) limita el uso del algoritmo a problemas donde exista una relación cronológica; b) requiere que la señal original se normalice en el rango $[\min, \max]$ y la señal portadora en el rango $[\min - \delta, \max + \delta]$ para garantizar que se produzcan intersecciones en todos los casos posibles; c) aumenta el coste computacional del método al tener que realizar una interpolación entre cada dos puntos consecutivos para después calcular la intersección entre dos rectas; y d) el algoritmo presenta ciertos inconvenientes para poder ser implementado en tiempo real al requerir de valores de dos instantes consecutivos.

2.2. Diseño de la extensión del algoritmo PWM.

La nueva versión del algoritmo de codificación y decodificación basado en PWM pretende solventar todas las limitaciones citadas en el apartado anterior. Así, esta nueva versión durante la codificación no requiere de dos puntos consecutivos de la serie temporal, sino que calcula directamente la intersección entre la señal portadora y cada uno de los valores reales que conforma la serie temporal.

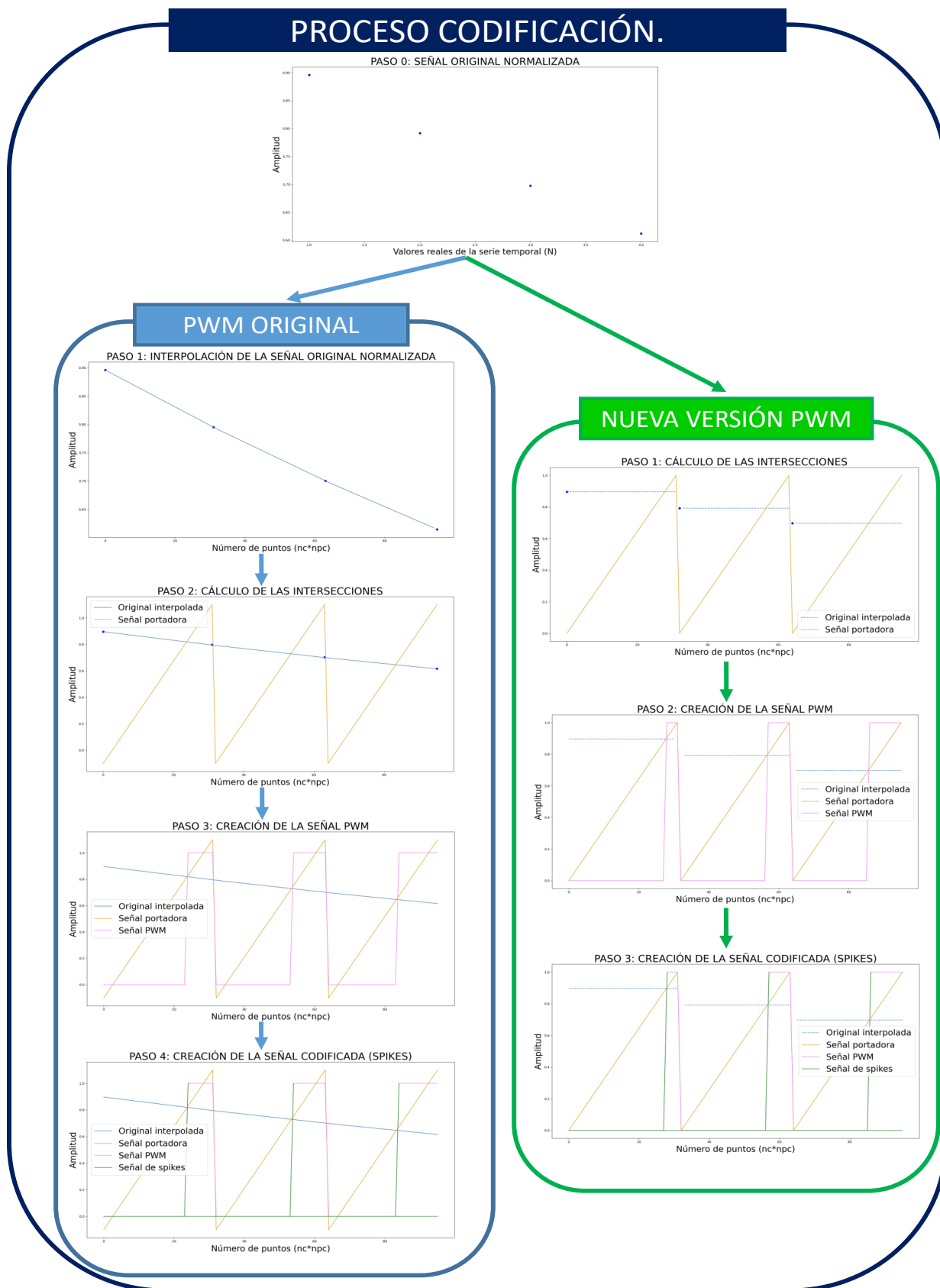


Figura 1: Comparación funcionamientos de la versión original del PWM vs. versión nueva.

La premisa en la que se basa la extensión del algoritmo basado en PWM es que, cuando no dispones de dos instantes temporales consecutivos, es posible codificar los valores analógicos (eje y) en el dominio temporal (eje x) simplemente mediante la localización del instante en el que la señal portadora pasa por el valor original. Esto es posible debido a que la señal portadora que comúnmente se utiliza es la señal diente de sierra, la cual para cada instante de tiempo se puede definir mediante la ecuación de una recta ($y = m \cdot x + b$). De esta ecuación la *pendiente de la recta* (m) y el *término independiente* (b) son totalmente conocidos ya que el diente de sierra se normaliza dentro del rango [min,max], el cual normalmente se representa mediante el rango [0,1]. En este sentido, la ecuación general de cada diente de sierra se puede definir mediante: $y = \frac{1}{npc-1} \cdot x$.

Conocida la ecuación que define la señal portadora en cada instante de tiempo y partiendo de la premisa citada anteriormente, la localización de los *spikes* en la fase de codificación se puede calcular directamente introduciendo en la ecuación del diente de sierra el valor real de la serie temporal normalizado en el rango [min, max] y despejando la variable x. Todo este proceso se puede visualizar en la Figura 1. Destacar que a diferencia del algoritmo original, en esta nueva versión tanto la señal original como la portadora se pueden normalizar en el mismo rango ya que siempre va a haber un punto de intersección.

Esta nueva versión del algoritmo de codificación-decodificación basado en PWM presenta importantes ventajas comparándola con su predecesora: a) el hecho de sustituir tanto la interpolación como el cálculo de intersecciones entre rectas por el cálculo de una recta que pasa por un punto, permite disminuir el coste computacional del método; b) la codificación de cada *spike* depende exclusivamente del valor real de un único instante de la serie temporal, lo cual hace que esta nueva versión del algoritmo sea fácilmente implementable en tiempo real; c) el punto codificado sólo depende de su correspondiente valor real, eliminando así cualquier relación cronológica con otros puntos y haciendo que la nueva versión del algoritmo pueda ser empleada no sólo con series temporal sino con imágenes u otro tipo de aplicaciones; y d) al no requerir del punto siguiente para interpolar, se pueden conseguir tantas intersecciones como valores reales tenga la serie.

3. Validación de la extensión del algoritmo.

La validación de la nueva propuesta de extensión del algoritmo de codificación-decodificación basado en PWM se va a realizar desde dos perspectivas. Por un lado, se va a comparar la precisión de ambas versiones del algoritmo, codificando y decodificando dos series temporales distintas. Por otro lado, se va a demostrar que la nueva versión del algoritmo se puede aplicar en la codificación y decodificación de imágenes.

3.1. Series temporales.

Se van a emplear dos series temporales diferentes para comparar los procesos de codificación y decodificación de la versión original del algoritmo PWM respecto a la extensión propuesta. Para cuantificar la variaciones existentes entre la señal original y las decodificadas por las versiones del algoritmo se va a

emplear las métricas de Error Cuadrático Medio (*Mean Square Error*, MSE):

$$MSE = \sum_{i=1}^{nc \cdot npc} (y_i - \hat{y}_i)^2 \quad (1)$$

donde y_i es el valor de cada muestra de la serie temporal original e \hat{y}_i es valor de cada muestra de la señal decodificada con la versión original del algoritmo PWM o con la nueva versión.

Además, se van a emplear los siguientes valores de hiperparámetros:

- $nc = N-1$ (versión original) y $nc = N$ (nueva versión). En Arriandiaga et al. (2020) se definió que el valor óptimo de este hiperparámetro es igual al número de valores reales de la serie temporal (N) menos 1; en cambio, en el nuevo algoritmo se está en disposición de tener tantas intersecciones como valores tenga la serie temporal.
- $npc = [32, 64, 128]$. Para verificar cómo influye la resolución en ambas versiones, se va a variar el parámetro npc entre los valores 32, 64 y 128.

La primera serie temporal es una onda senoidal de frecuencia $f_o = 100Hz$, amplitud $A = 3$, frecuencia de muestreo $f_s = 60 \cdot f_o$ y número de ciclos 20. Estos parámetros dan lugar a una serie temporal de 1200 valores.

En la Tabla 1 se muestran los errores MSE y el coste computacional (CC) en segundos obtenidos con esta serie temporal. Se puede comprobar que el CC es inferior para todos los valores de npc estudiados con la nueva versión y que al aumentar la resolución del algoritmo (npc) los errores MSE disminuyen tanto para el caso de la versión original del algoritmo PWM como para la versión nueva. También se puede observar que los errores de la versión original son inferiores a los ofrecidos por la nueva versión. Así, la Figura 2a muestra la evolución tanto de la señal original como de las señales decodificadas con ambas versiones del algoritmo PWM para un valor de npc igual a 32.

Tabla 1: Métricas obtenidas para serie temporal de la señal senoidal.

		SENOIDAL	
		Versión PWM original	Versión PWM nuevo
Npc = 32	MSE	0.00045	0.00051
	CC (s)	0.3047	0.1106
Npc = 64	MSE	0.00011	0.00028
	CC (s)	0.6018	0.2158
Npc = 128	MSE	0.00003	0.00022
	CC (s)	1.199	0.4271

La segunda serie temporal se ha extraído de una base de datos UCI que contiene grabaciones de voz de personas que se llevaron cabo en *Carnegie Mellon University*. Esta base de datos es conocida como ARCTIC, la cual está disponible en Black (2003) y explicada en Kominek and Black (2003). Cada voz está grabada a una frecuencia de 32kHz. En este trabajo se ha seleccionado el fichero llamado "*arctic.a0001.wav*" dentro de la carpeta "*US English bdl (made)*". Dado el gran volumen de datos disponibles, se ha seleccionado el rango de datos comprendido entre la muestra 4985 y la muestra 7425 (2440 valores discretos).

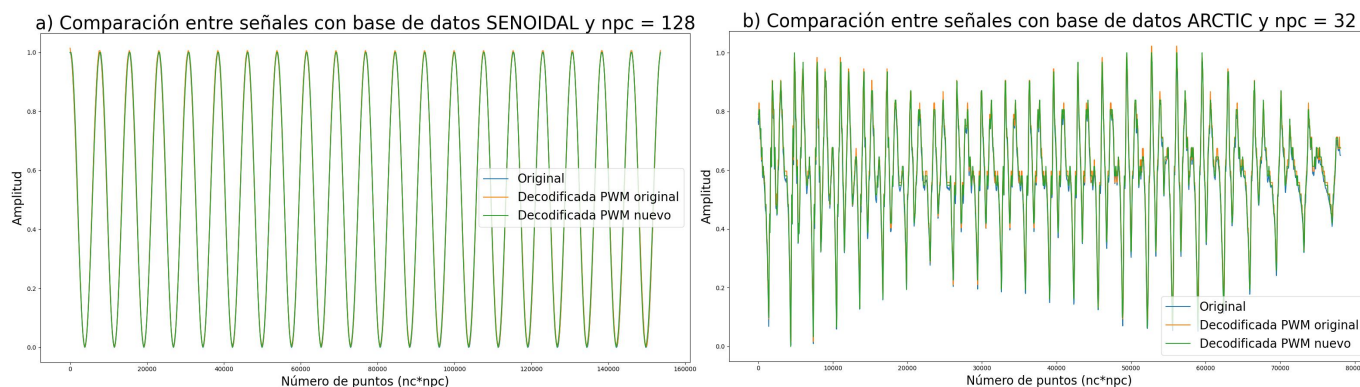


Figura 2: Comparación señal original con las señales decodificadas por el algoritmo PWM con la versión original y la nueva versión.

Así, la Tabla 2 muestra los errores y el CC obtenidos con la serie temporal seleccionada. Al igual que en el caso anterior, se demuestra que el CC es inferior con la nueva versión para cualquier valor de npc y al aumentar la resolución del algoritmo los errores MSE disminuyen tanto para el caso de la versión original del algoritmo PWM como para la versión nueva. Por otro lado, también se puede observar que con npc igual a 32 los errores MSE obtenidos son bastante similares para ambas versiones; sin embargo, al aumentar npc la versión original del algoritmo PWM ofrece menores errores. Por último, en la Figura 2b se puede ver la evolución de todas las señales, empleando un valor de npc igual a 128 para ambas versiones del algoritmo PWM.

Tabla 2: Métricas obtenidas para la serie temporal de la base de datos ARCTIC.

		ARCTIC	
		Versión PWM original	Versión PWM nuevo
Npc = 32	MSE	0.00048	0.00048
	CC (s)	0.6387	0.2411
Npc = 64	MSE	0.00011	0.00022
	CC (s)	1.2605	0.4706
Npc = 128	MSE	0.00003	0.00017
	CC (s)	2.5092	0.9422

3.2. Imágenes.

Dado que las SNN se utilizan ampliamente en el tratamiento y procesamiento de imágenes, en esta sección se valida la aplicación de la nueva versión del algoritmo en este campo. En este sentido, a continuación se codifica y decodifica una misma imagen con diferentes valores de resolución. Más concretamente, los hiperparámetros considerados son:

- $nc = N$, siendo N el número total de píxeles de la imagen.
- $npc = [2, 4, 8, 16, 32, 64, 128, 256]$.

Para comparar las diferentes imágenes se calcula el MSE, que en el caso de las imágenes se basa en comparar los valores píxel a píxel según la Ecuación 1.

En la Figura 3 se muestra el resultado de la codificación y decodificación de las imágenes para todos los valores de npc mencionados anteriormente. Así, se puede observar que al

aumentar la resolución del algoritmo (npc), la calidad de las imágenes mejora sustancialmente, lo cual también se corrobora con los valores MSE recogidos en la Tabla 3. El hecho que para npc igual a 256 se consiga un error nulo quiere decir que el proceso de codificación y decodificación con la nueva versión del algoritmo basado en PWM es adecuado para la codificación y decodificación de imágenes, ya que la resolución de un píxel está en el rango $[0-255]$.

Tabla 3: MSE obtenido variando el npc en la codificación y decodificación de la imagen seleccionada.

npc	MSE	npc	MSE
2	25873.17	32	22.43
4	2476.68	64	5.46
8	450.72	128	1.19
16	88.12	256	0

4. Discusión.

La nueva versión del algoritmo de codificación-decodificación basado en PWM ha demostrado que permite codificar y decodificar tanto series temporales como imágenes con resultados muy satisfactorios.

En cuanto a la codificación y decodificación de series temporales la nueva versión del algoritmo ha demostrado ser más eficiente computacionalmente que la versión original. Asimismo, si bien es cierto que la nueva versión del algoritmo no consigue métricas de error tan bajas como la versión original, la diferencia en dichas métricas no es muy significativa. Además, analizando con más profundidad los resultados obtenidos, se puede destacar que la nueva versión del algoritmo se adapta mejor a los cambios de pendiente producidos en la señal; sin embargo, es peor en los tramos más rectos, tal y como se puede ver en la Figura 2. Esto se debe a que el proceso de interpolación de la versión original hace un ajuste más fino al localizar el punto de intersección en los tramos rectos; sin embargo, en los cambios de pendiente no localiza el valor de pico con precisión. Por otro lado, con la nueva versión esta información no se pierde porque el valor real del pico siempre va a estar representado en la codificación.

Respecto al tratamiento de imágenes, la nueva versión del algoritmo ha demostrado ser capaz de codificar y decodificar

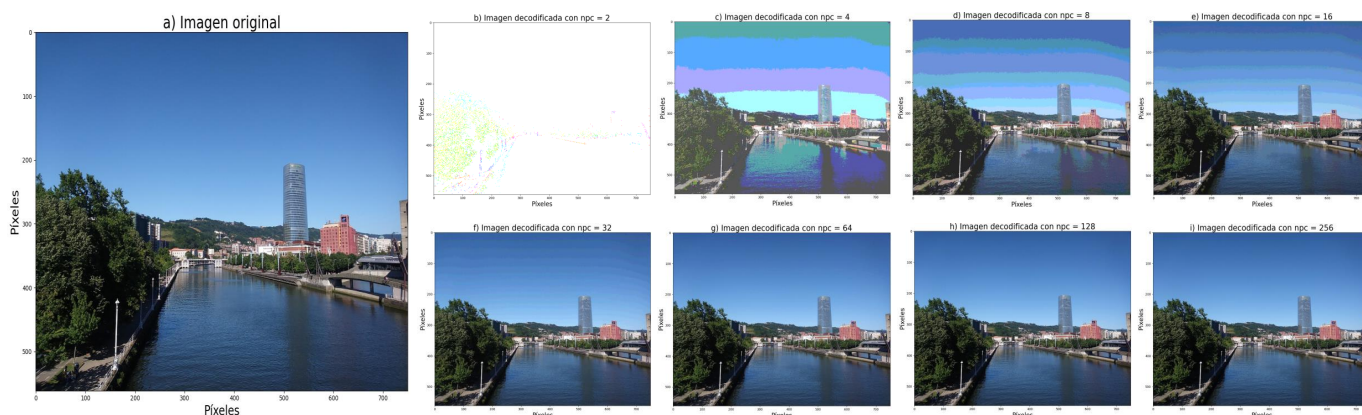


Figura 3: Diferencias en la codificación y decodificación de imágenes variando el npc.

imágenes con gran precisión, permitiendo su uso en aplicaciones junto a las SNN.

Por todo ello, la nueva versión propuesta del algoritmo permite: a) reducir el coste computacional del método al simplificar las operaciones en las que está basado el propio algoritmo; b) tener un algoritmo de codificación susceptible de ser implementado más eficientemente en tiempo real; y c) ampliar su uso a otros campos de aplicación, a parte de las series temporales, donde no existen relaciones cronológicas entre los datos empleados. En este sentido, el usuario puede decidir qué versión utilizar dependiendo del campo de aplicación donde lo quiera implementar.

5. Conclusiones

En este trabajo se presenta una nueva versión del algoritmo de codificación-decodificación basado en PWM presentado en Lucas et al. (2021) Arriandiaga et al. (2020). Para validar esta nueva versión del algoritmo se ha comparado con su versión anterior en la codificación y decodificación de series temporales y, además, se ha validado en un nuevo campo de aplicación como es el procesamiento de imágenes.

Los resultados reflejan que la nueva versión del algoritmo consigue resultados muy satisfactorios en ambos ámbitos, siendo la versión original ligeramente más precisa en el campo de las series temporales. Sin embargo, la posibilidad de poder aplicar a campos donde no existan relaciones cronológicas así como su mayor simplicidad para ser implementado en tiempo real supone una ventaja para favorecer la usabilidad de este algoritmo de codificación y decodificación.

Agradecimientos

Este trabajo ha sido financiado por el Departamento de Educación del Gobierno Vasco (proyectos ref. PIBA_2020_1_0008 y ref. IT1726-22), el FEDER/Ministerio de Ciencia e Innovación - Agencia Estatal de Investigación/Proyecto (ref. PID2020-112667RB-I00) y por el Programa Euskampus de la Fundación Misiones Euskampus (proyecto NEURO TIP).

Referencias

- Arriandiaga, A., Portillo, E., Espinosa-Ramos, J. I., Kasabov, N. K., 2020. Pulsewidth Modulation-Based Algorithm for Spike Phase Encoding and Decoding of Time-Dependent Analog Data. *IEEE Transactions on Neural Networks and Learning Systems* 31 (10), 3920–3931. DOI: 10.1109/TNNLS.2019.2947380
- Black, A. W., 2003. Cmu.arctic speech synthesis databases. <http://festvox.org/cmu.arctic/>, accessed: 2023-02-07.
- Brusca, S., Capizzi, G., Lo Sciuto, G., Susi, G., jul 2019. A new design methodology to predict wind farm energy production by means of a spiking neural network-based system. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields* 32 (4). DOI: 10.1002/JNM.2267
- Chen, T., Sun, G., Wei, Z., Li, H., Cheung, K. W., Sun, Y., 2016. Photovoltaic system power generation forecasting based on spiking neural network. *Proceedings of the 2015 Chinese Intelligent Systems Conference* 359, 573–581. DOI: 10.1007/978-3-662-48386-2_59/TABLES/3
- Gerstner, W., Kistler, W. M., aug 2002. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press. DOI: 10.1017/CB09780511815706
- Hong, C., Wei, X., Wang, J., Deng, B., Yu, H., Che, Y., apr 2020. Training Spiking Neural Networks for Cognitive Tasks: A Versatile Framework Compatible with Various Temporal Codes. *IEEE Transactions on Neural Networks and Learning Systems* 31 (4), 1285–1296. DOI: 10.1109/TNNLS.2019.2919662
- Kominek, J., Black, A. W., 2003. CMU ARCTIC databases for speech synthesis.
- Laña, I., Capecci, E., Del Ser, J., Lobo, J. L., Kasabov, N., 2018. Road traffic forecasting using neucube and dynamic evolving spiking neural networks. *International Symposium on Intelligent and Distributed Computing* 798, 192–203. DOI: 10.1007/978-3-319-99626-4_17
- Lopes-dos Santos, V., Panzeri, S., Kayser, C., Diamond, M. E., Quian Quiroga, R., 2015. Extracting information in spike time patterns with wavelets and information theory. *Journal of Neurophysiology* 113 (3), 1015–1033. DOI: 10.1152/JN.00380.2014
- Lucas, S., Arriandiaga, A., Portillo, E., Zubizarreta, A., Cabanes, I., aug 2021. Compresión de datos de tipo real basada en un novedoso algoritmo de codificación para redes neuronales de impulsos. *Proceedings of the XLIII Jornadas de Automática*, 175–182. DOI: 10.17979/SPUDC.9788497498043.175
- Xu, Y., Tang, H., Xing, J., Li, H., feb 2017. Spike trains encoding and threshold rescaling method for deep spiking neural networks. *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–6. DOI: 10.1109/SSCI.2017.8285427