

Spiking Neural Networks

María Luz Stewart Harris

November 26, 2025



Instituto
Balseiro

¿Qué es una SNN?	3
Neuronas de una SNN	6
Codificación de información en una SNN	8
¿Por qué usar SNNs?	10
SNNs aplicadas a series temporales	13
Hardware neuromórfico	14
Caso de uso: cámaras de eventos.	14
¿Cómo se entrenan las SNNs?	18

¿Qué es una SNN?

¿Por qué usar SNNs?

¿Cómo se entrenan las SNNs?

Spiking Neural Networks

Definición (Spiking Neural Networks / SNN)

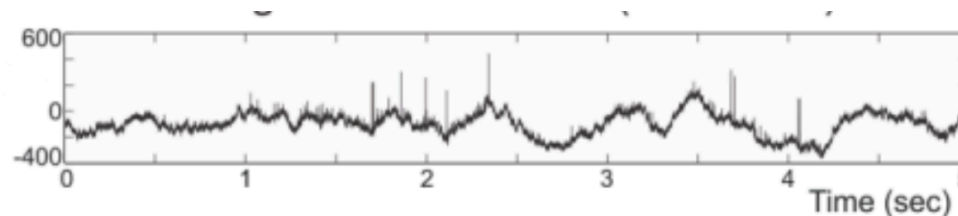
Redes neuronales artificiales en el que el único intercambio que ocurre entre neuronas es el de pulsos de igual amplitud en diferentes instantes. [1]

Spiking Neural Networks

Definición (Spiking Neural Networks / SNN)

Redes neuronales artificiales en el que el único intercambio que ocurre entre neuronas es el de pulsos de igual amplitud en diferentes instantes. [1]

- *Inspiración de redes biológicas*: la comunicación entre neuronas biológicas sucede a través de señales eléctricas que contienen pulsos:



Spiking Neural Networks

- ▶ En una SNN, la información se codifica exclusivamente por la posición de los pulsos transmitidos.
- ▶ La amplitud y la forma de los pulsos **no** aportan más información.

Neuronas de una SNN

Características necesarias de las neuronas de una SNN:

¹Aplica a neuronas individuales, independientemente de si la arquitectura de la red implementa memoria a través de recurrencia.

Neuronas de una SNN

- ▶ **Memoria**: Su salida en $t = t_0$ depende de la entrada en $t \leq t_0$.¹

Características necesarias de las neuronas de una SNN:

¹Aplica a neuronas individuales, independientemente de si la arquitectura de la red implementa memoria a través de recurrencia.

Neuronas de una SNN

Características necesarias de las neuronas de una SNN:

- ▶ **Memoria**: Su salida en $t = t_0$ depende de la entrada en $t \leq t_0$.¹
- ▶ **Event-driven**: no transmiten información todo el tiempo, solamente cuando ocurrió un evento.

¹Aplica a neuronas individuales, independientemente de si la arquitectura de la red implementa memoria a través de recurrencia.

Neuronas de una SNN

Características necesarias de las neuronas de una SNN:

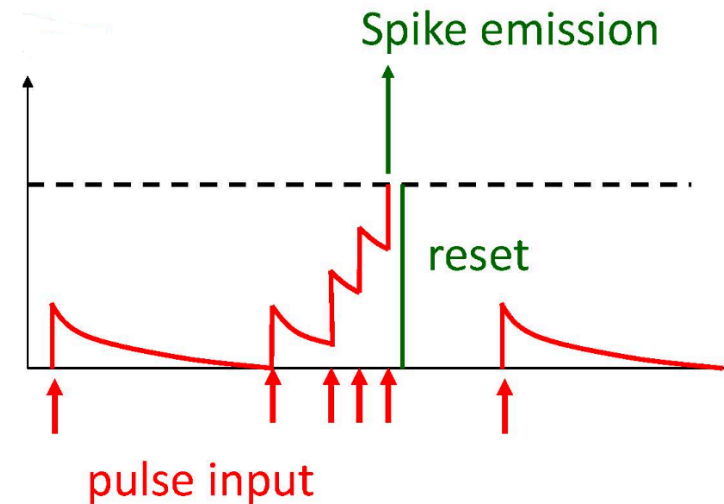
- ▶ **Memoria**: Su salida en $t = t_0$ depende de la entrada en $t \leq t_0$.¹
- ▶ **Event-driven**: no transmiten información todo el tiempo, solamente cuando ocurrió un evento.
- ▶ **Async**: generan un disparo apenas detectan un evento, sin tener que esperar a otras neuronas.

¹Aplica a neuronas individuales, independientemente de si la arquitectura de la red implementa memoria a través de recurrencia.

Leaky Integrate-and-Fire

- Existen muchos modelos matemáticos de neuronas apropiados para una SNN, entre ellos el modelo *Leaky Integrate-and-Fire* (LIF).

$$\begin{cases} \dot{s}(t) = \frac{1}{\tau_s} [s_{\text{rest}} - s(t)] + x(t) \\ y(t) = \sum_i \delta(t - t_{f_i}) \\ s(t_{f_i}^+) = s_{\text{rest}} \end{cases}$$

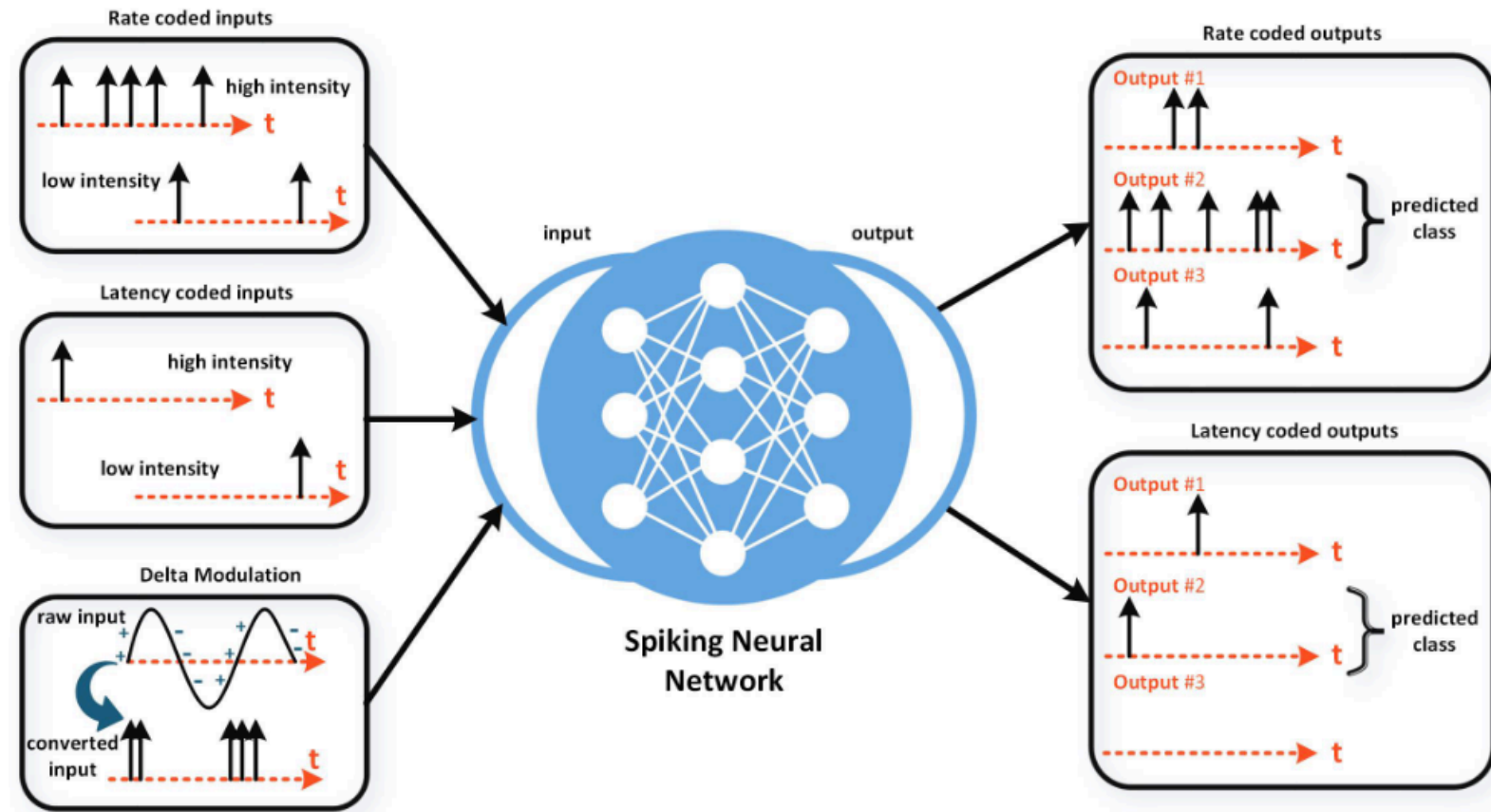


Codificación de información en una SNN

Métodos de codificación de estímulo de entrada en pulsos:

- ▶ ***Rate encoding***: Convierte intensidad del estímulo de entrada en una tasa de disparos.
- ▶ ***Temporal encoding***: Convierte intensidad del estímulo de entrada en un tiempo de disparo.
- ▶ ***Delta modulation***: Convierte *cambios* de intensidad del estímulo de entrada en disparos

Codificación de información en una SNN



[2]

¿Qué es una SNN?

¿Por qué usar SNNs?

¿Cómo se entrenan las SNNs?

¿Por qué usar SNNs?

- ▶ ***Eficiencia energética***: en muchos casos, el consumo de potencia es mucho menor principalmente debido a que:
 - son *event-driven*: las neuronas solo transmiten información (i.e.: un pulso) cuando ocurre un evento.
 - las señales de salida son poco densas (*sparse*) debido a que, para cada neurona, la mayor parte del tiempo no hay eventos.¹

¹Esto depende del entrenamiento, pero es posible incluir la cantidad de disparos total de la red como factor en la función de costo para regular el consumo de potencia total de la red.

¿Por qué usar SNNs?


- ▶ **Eficiencia energética:** en muchos casos, el consumo de potencia es mucho menor principalmente debido a que:
 - son *event-driven*: las neuronas solo transmiten información (i.e.: un pulso) cuando ocurre un evento.
 - las señales de salida son poco densas (*sparse*) debido a que, para cada neurona, la mayor parte del tiempo no hay eventos.¹
- ▶ **Baja latencia:**

¹Esto depende del entrenamiento, pero es posible incluir la cantidad de disparos total de la red como factor en la función de costo para regular el consumo de potencia total de la red.

¿Por qué usar SNNs?

- ▶ *Apropiadas para procesamiento series temporales:*
 - Como las neuronas tienen memoria su entrada en el pasado, no hace falta una arquitectura recurrente para procesar temporales.
 -

¿Por qué usar SNNs?

- ▶ *Apropiadas para procesamiento series temporales:*
 - Como las neuronas tienen memoria su entrada en el pasado, no hace falta una arquitectura recurrente para procesar temporales.
 - 
- ▶ *Similitud a redes neuronales biológicas:* son más cercanas a las redes neuronales biológicas, lo que puede ser útil para modelar y estudiar su comportamiento.

DNNs vs. SNNs

Comparación entre DNNs y SNNs para aplicación de computer vision: [3]

	DNNs	SNNs
Procesamiento de datos	Basado en frames	Basado en eventos
Latencia	Alta	Baja
Diferenciable	Sí	No
Complejidad de neurona	Baja	Alta
Memoria a corto plazo	Nivel red	Nivel red y neuronal
Eficiencia energética ¹	Baja	Alta

¹Considerando hardware neuromórfico específico para cada caso.

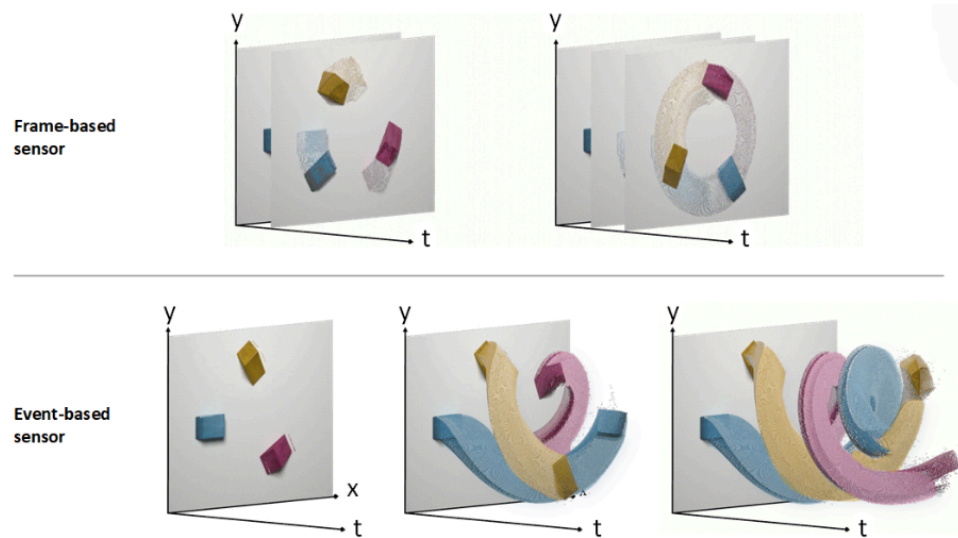
Hardware neuromórfico

Hardware neuromorfico, por que y que existe comercialmente / semicomercialmente

Caso de uso: cámaras de eventos.

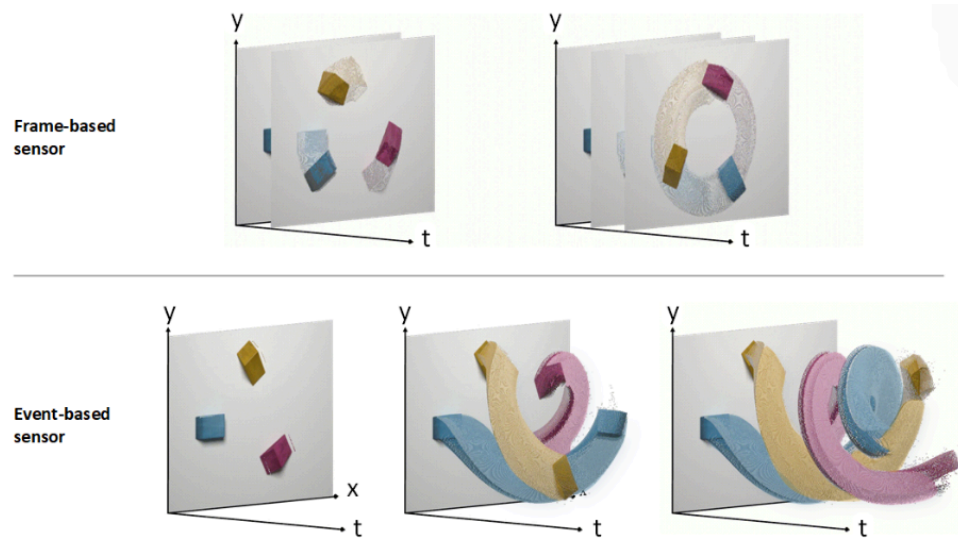
Caso de uso: cámaras de eventos.

- Cámaras inspiradas en sistemas visuales biológicos:

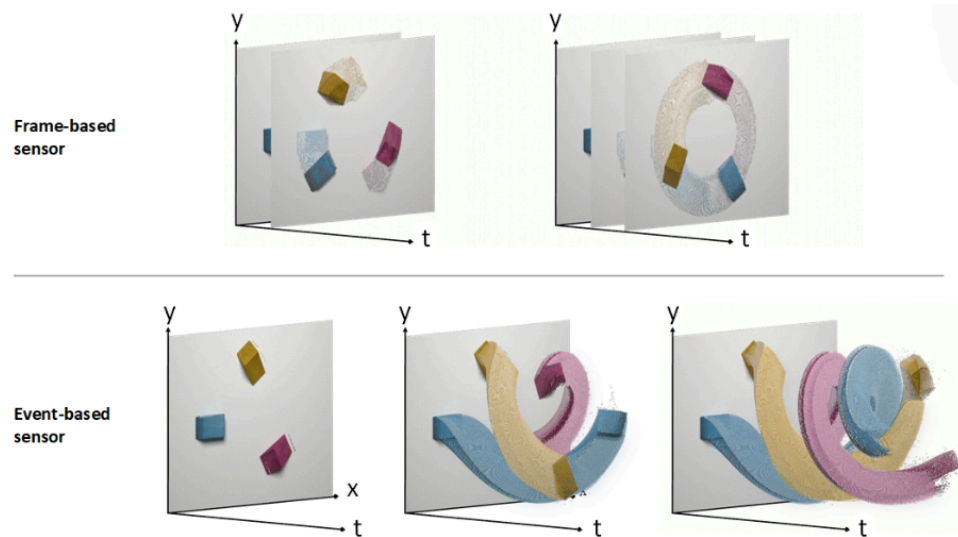


Caso de uso: cámaras de eventos.

- ▶ Cámaras inspiradas en sistemas visuales biológicos:
 - Mayor sensibilidad a grandes cambios de intensidad que a intensidad constante.

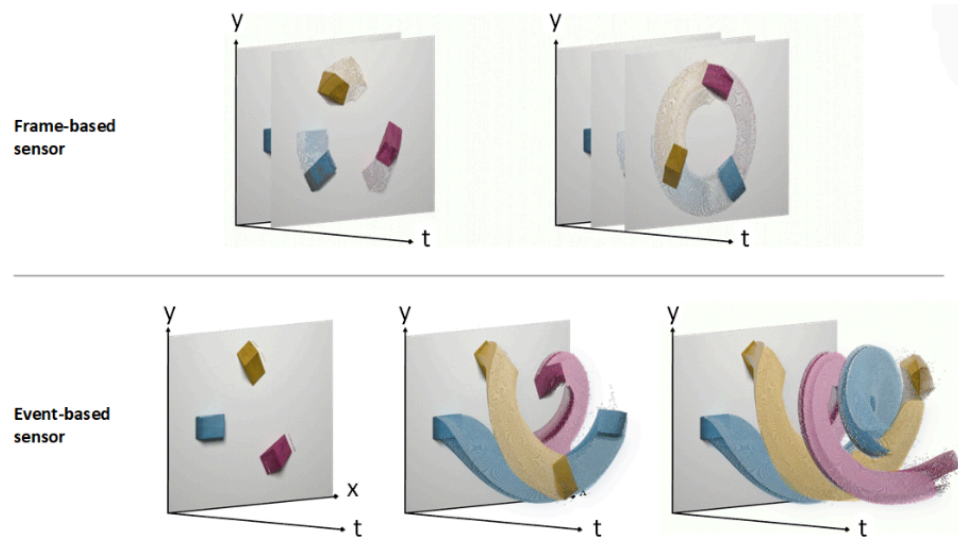


Caso de uso: cámaras de eventos.



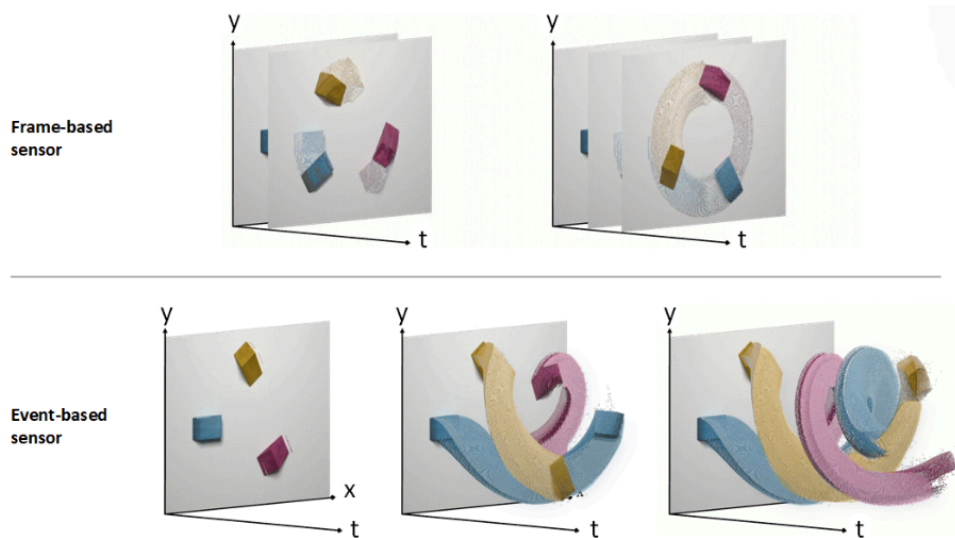
- ▶ Cámaras inspiradas en sistemas visuales biológicos:
 - Mayor sensibilidad a grandes cambios de intensidad que a intensidad constante.
 - *Event-driven*: solo se guarda la información de eventos (cambios). Donde no hay cambios no se guarda información.

Caso de uso: cámaras de eventos.



- ▶ Cámaras inspiradas en sistemas visuales biológicos:
 - Mayor sensibilidad a grandes cambios de intensidad que a intensidad constante.
 - *Event-driven*: solo se guarda la información de eventos (cambios). Donde no hay cambios no se guarda información.

Caso de uso: cámaras de eventos.



- ▶ La salida es un arreglo de estructuras iguales con 3 datos (no se guardan frames completos!):
 - Posición del evento
 - Timestamp del evento:
 - Polaridad del evento
- ⇒ Comparada con una cámara de frames convencional, tiene:
 - ▶ Menos volumen de datos
 - ▶ Más detalle del movimiento

Caso de uso: cámaras de eventos.

tabla de comparacion entre frame based y event based camaras.

Combinacion con camara frame based

- ▶ ventajas:
 - muy baja latencia, muy bajo motion blur, muy baja redundancia de datos

¿Qué es una SNN?

¿Por qué usar SNNs?

¿Cómo se entrenan las SNNs?

¿Cómo se entrenan las SNNs?

supervisado vs no supervisado

supervisado: backpropagation y gradient descent, desafío de la no derivabilidad

¿Cómo se entrenan las SNNs?

Ejemplo snntorch. El de las estrellas quizás?

Muchas gracias! 🧠

Leaky Integrate-and-Fire

Definición (Modelo Leaky Integrate-and-Fire (LIF))

$$\begin{cases} \dot{s}(t) = \frac{1}{\tau_s} [s_{\text{rest}} - s(t)] + x(t) \\ y(t) = \sum_i^{\infty} \delta(t - t_{f_i}) \\ s(t_{f_i}^+) = s_{\text{rest}} \end{cases} \quad (1)$$

- ▶ t_{f_i} : t tal que $s(t) \geq s_{th}$,
- ▶ $x(t)$: entrada (pulsos)
- ▶ $s(t)$: estado de la neurona
- ▶ $y(t)$: tren de deltas (disparos)

Referencias

Referencias

- W. Maass, “Networks of spiking neurons: The third generation of neural network models,” *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, Dec. 1997, doi: [10.1016/S0893-6080\(97\)00011-7](https://doi.org/10.1016/S0893-6080(97)00011-7).
- [1]
- J. K. Eshraghian *et al.*, “Training Spiking Neural Networks Using Lessons From Deep Learning,” no. arXiv:2109.12894, Aug. 2023, doi: [10.48550/arXiv.2109.12894](https://doi.org/10.48550/arXiv.2109.12894).
- [2]
- H. Hendy and C. Merkel, “Review of spike-based neuromorphic computing for brain-inspired vision: biology, algorithms, and hardware,” *Journal of Electronic Imaging*, vol. 31, no. 1, p. 10901, Jan. 2022, doi: [10.1117/1.JEI.31.1.010901](https://doi.org/10.1117/1.JEI.31.1.010901).
- [3]