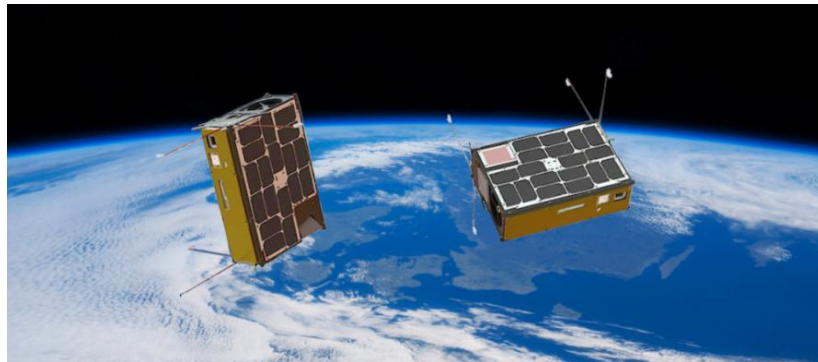
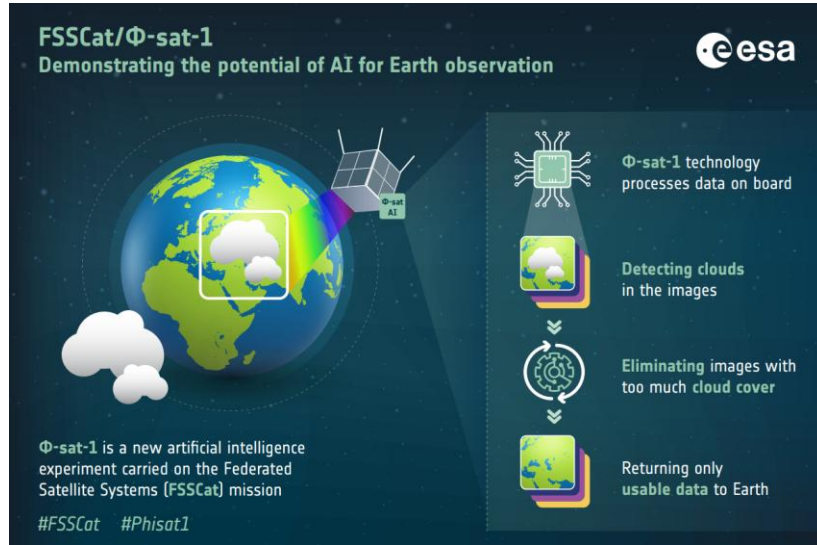


Cátedra IA3 – PT7

Mejora de las capacidades de procesamiento a bordo mediante algoritmos de Aprendizaje Automático

Introducción





- Ahorro de ancho de banda en el down-link a Tierra
- Incremento de la capacidad de procesamiento a bordo
- Menor latencia o uso de recursos frente a algoritmos clásicos
- Modelos simples
- Impredecibilidad ante nuevos escenarios
- Riesgo

Introducción

Research paper

End-to-end deep learning pipeline for on-board extraterrestrial rock segmentation

Daniel Marek ^{a b}, Jakub Nalepa ^{a b}  

Few-shot satellite image classification for bringing deep learning on board OPS-SAT



Ramez Shendy , Jakub Nalepa  

Robust Machine Learning Systems For Dependable Space Applications

Publisher: **IEEE** [Cite This](#) [PDF](#)

Nikolaos Panagiotopoulos ; Toni Lubiniecki ; Alen Turnwald ; Niklas Baldauf [All Authors](#)

Edge computing in space: Design of an FPGA architecture for thermal anomaly detection based on a machine learning approach

Carmen Misa Moreira , Carl Shneider , Andreas M. Hein 

Informational and extreme machine learning for onboard recognition system of ground objects

Publisher: **IEEE** [Cite This](#) [PDF](#)

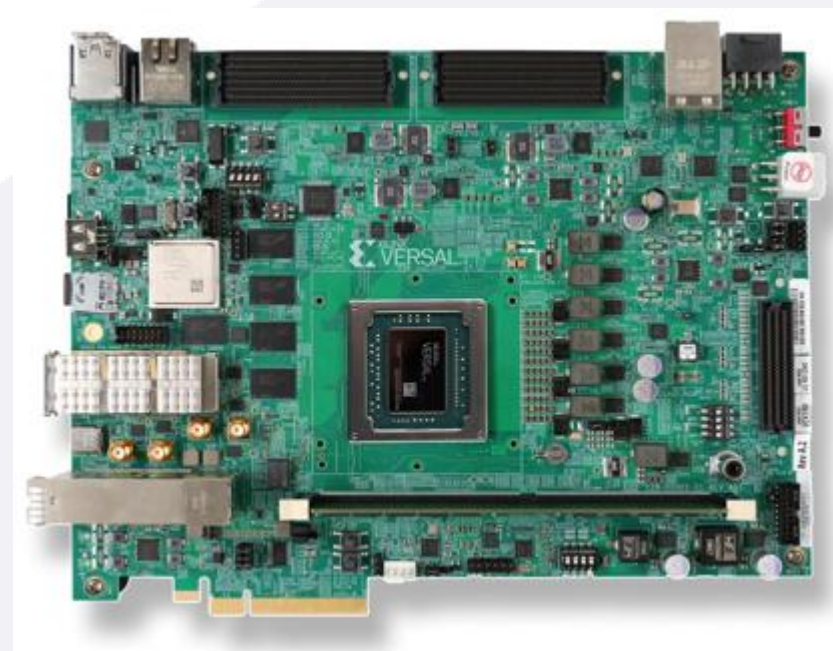
Olena Protsenko ; Taras Savchenko ; Mykyta Myronenko ; Oleksandr Prikhodchenko

[All Authors](#)

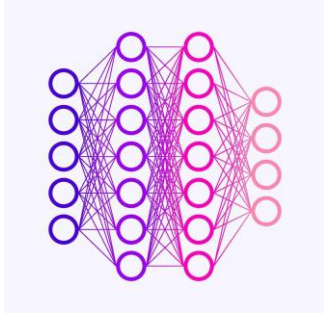
In-Orbit Artificial Intelligence and Machine Learning On-Board Processing Solutions for Space Applications : Edge-Based and Versal Space Reference Designs : First Design-In Experiences

Publisher: **IEEE** [Cite This](#) [PDF](#)

Rajan Bedi [All Authors](#)



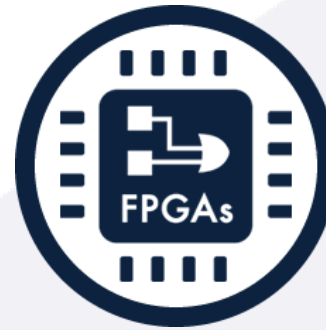
Objetivos



1. Diseño de modelos de Machine Learning eficientes para la resolución de tareas específicas a bordo.

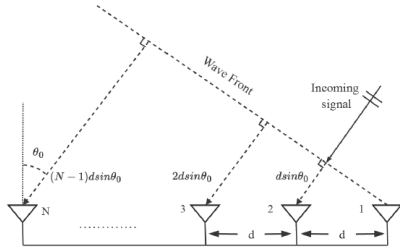


2. Entrenamiento de los algoritmos en software (Matlab, Python, Pytorch)

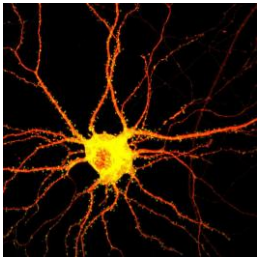


3. Implementación eficiente e inferencia en FPGA para procesamiento *on the edge*.

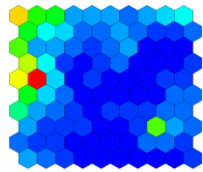
Proyectos



- **Beamforming Digital (DBF).** Comparativa entre algoritmo tradicional (MVDR) y FFNN en FPGA.



- **Spiking Neural Networks (SNN):**



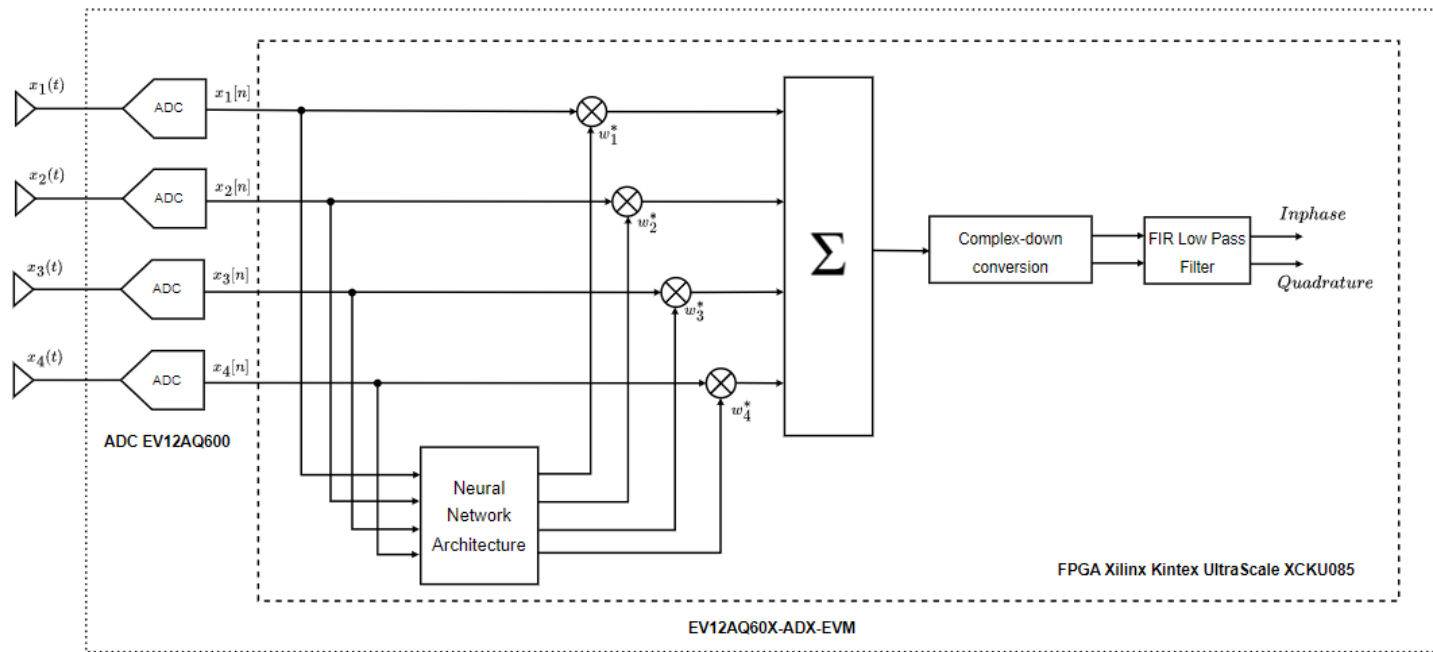
- **Clustering no supervisado.** Modelo SNN-SOM en FPGA



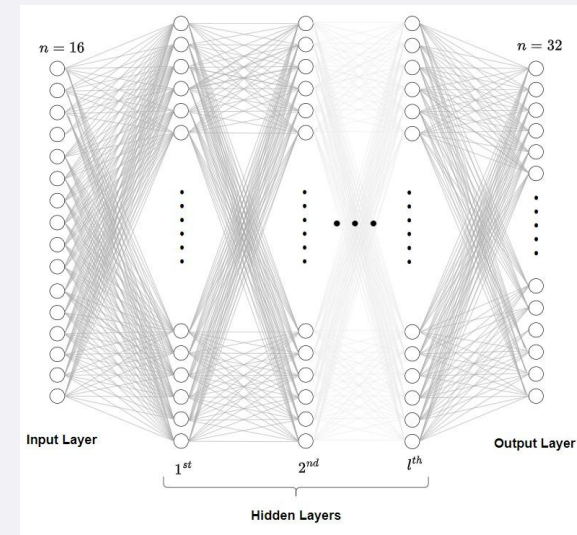
- **Clasificación de imágenes.** Modelo híbrido CNN-SNN

DBF mediante FFNN en FPGA

- Entrada: primera fila de la matriz de correlación
- Salida: coeficientes de beamforming



Red entrenada con datos sintéticos. Targets de entrenamiento calculados con algoritmo MVDR ideal



Resultados DBF FFNN

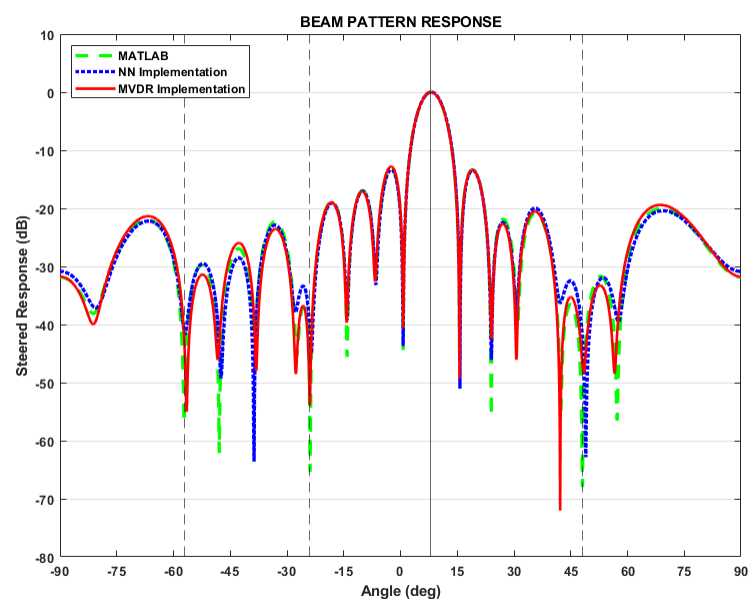
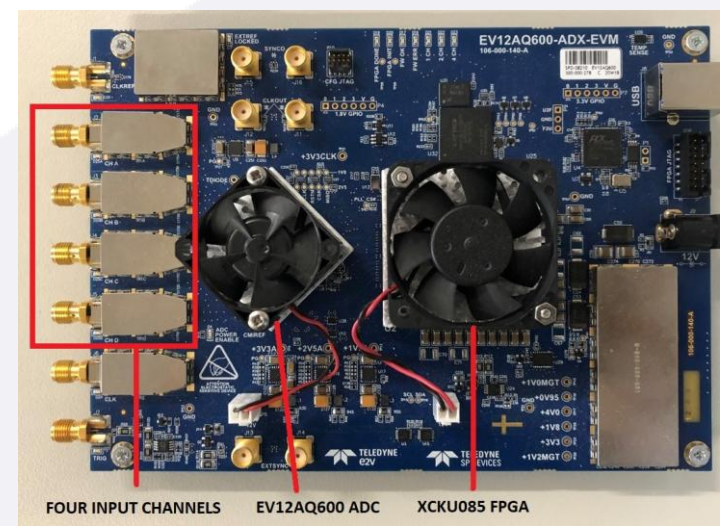


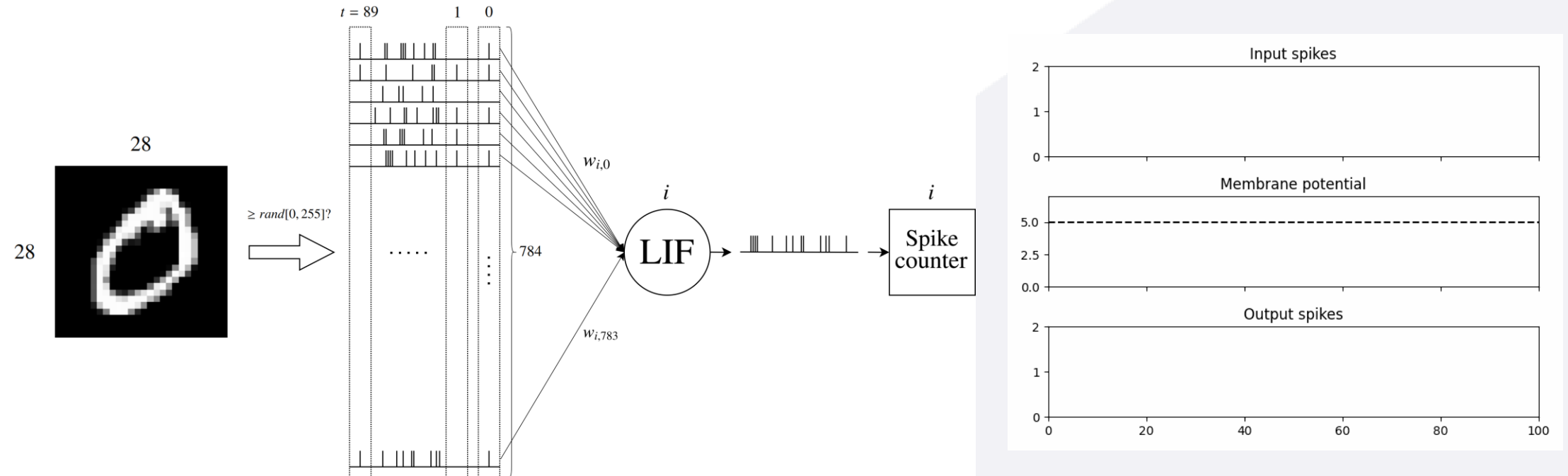
Table 22: Comparison Summary Between MVDR and FFNN (16 Antennas)

Metric	MVDR	FFNN	Note / Improvement
Average SNR difference	0 dB (ref.)	-0.187 dB	Minor degradation
Total latency	27.22 μ s	9.47 μ s	65% reduction
Power consumption	1.75 W	0.95 W	45% lower
DSP usage	3780	1728	>2 \times more efficient
CLB LUTs	429,520	143,053	67% reduction
Total multiplications	1938	928	52% fewer ops
Offline training time	N/A	24h (A100 GPU)	5 $\times 10^6$ samples



Clustering con SNN

- Aprendizaje no supervisado
- STDP (*Spike-timing dependent plasticity*) para modificar los pesos internos
- Clustering topológico: entradas parecidas activan neuronas cercanas (SOM)

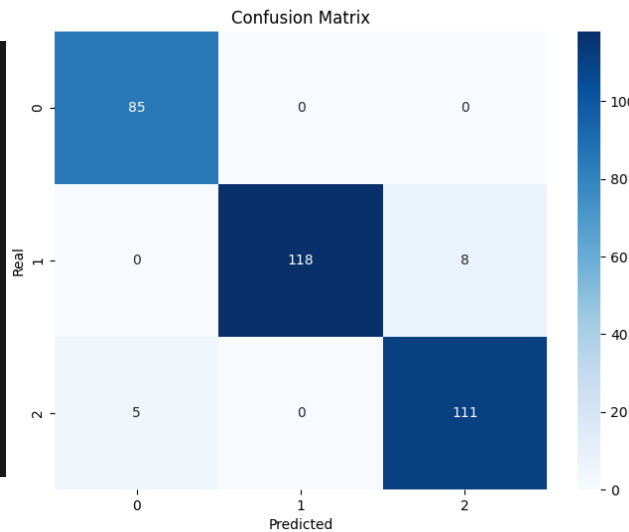


Clustering con SNN

Entrenamiento completado para TH = 44

```
[[2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [2 2 2 2 2 2 2 2 0 0 0 0 0]
 [1 1 1 1 1 1 2 2 0 0 0 0 0]
 [1 1 1 1 1 1 1 2 0 0 0 0 0]
 [1 1 1 1 1 1 1 1 0 1 0 1]
 [1 1 1 1 1 1 1 1 1 1 1 1]
 [1 1 1 1 1 1 1 1 1 1 1 1]
 [1 1 1 1 1 1 1 1 1 1 1 1]
 [1 1 1 1 1 1 1 1 1 1 1 1]
 [1 1 1 1 1 1 1 1 1 1 1 1]]
```

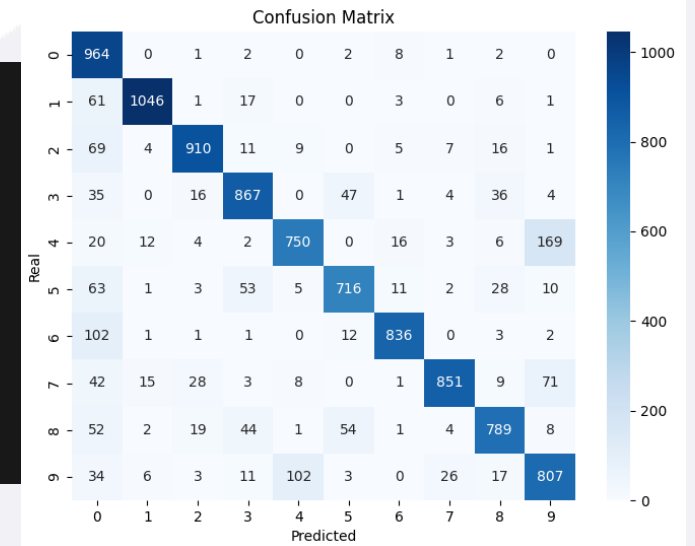
Total accuracy: 0.9602446483180428



Entrenamiento completado para T = 90

```
[[0 0 0 2 0 4 9 9 4 9 8 3 5 5]
 [0 0 0 0 3 9 7 4 4 4 2 5 5 5]
 [5 0 0 0 5 9 3 3 8 8 5 3 5 5]
 [5 6 6 5 8 3 3 3 3 9 1 8 8 8]
 [2 2 6 6 3 5 3 3 8 3 1 1 7 8]
 [2 2 2 6 3 3 3 3 1 1 1 1 1 1]
 [2 5 2 8 8 3 3 1 1 1 1 1 1 1]
 [6 2 4 2 2 3 3 1 1 1 1 1 1 1]
 [6 6 2 9 4 4 4 1 1 1 1 1 1 1]
 [6 6 2 4 9 4 9 9 9 1 1 1 1 1]
 [6 6 6 9 4 4 4 4 9 9 7 7 7 7]
 [6 6 4 4 9 4 4 4 9 7 9 7 7 7]
 [0 7 7 7 4 4 4 4 7 7 9 7 7 7]
 [7 7 7 7 4 4 4 4 7 9 7 9 7 2]]
```

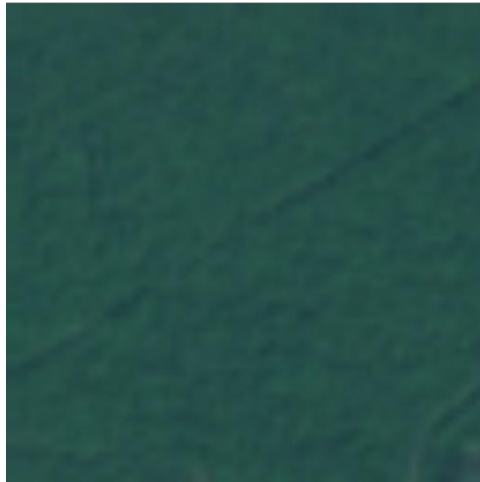
Total accuracy: 0.8536



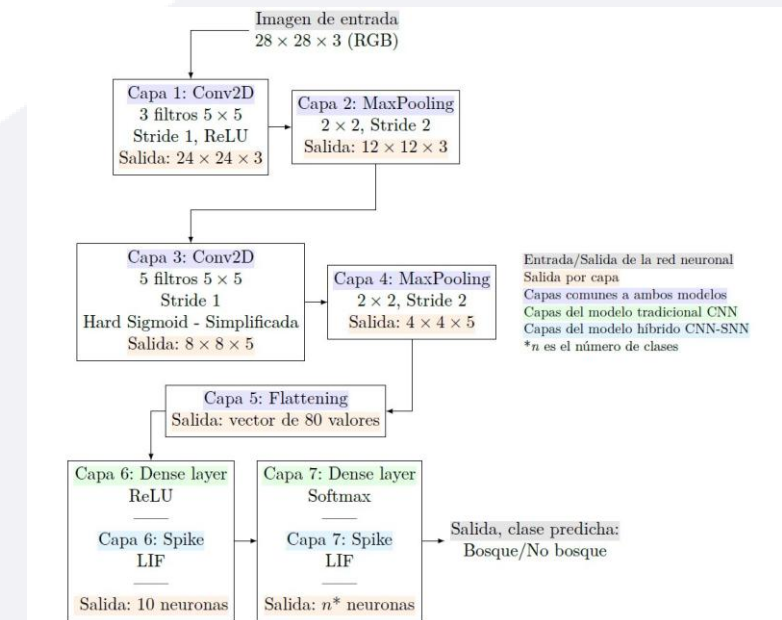
Clasificación de imágenes con SNN

- Distinción entre clases "Bosque" y "No bosque" en imágenes satelitales
- Red híbrida CNN-SNN
- Aprendizaje supervisado (backpropagation) con modelo CNN-FFNN y traslación de pesos a modelo híbrido para la inferencia

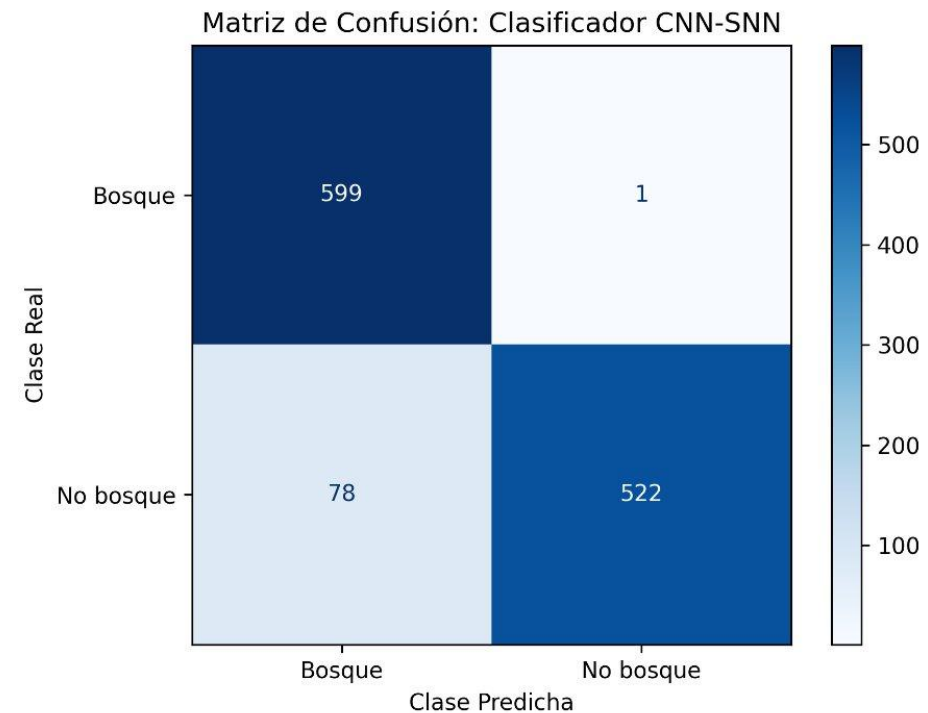
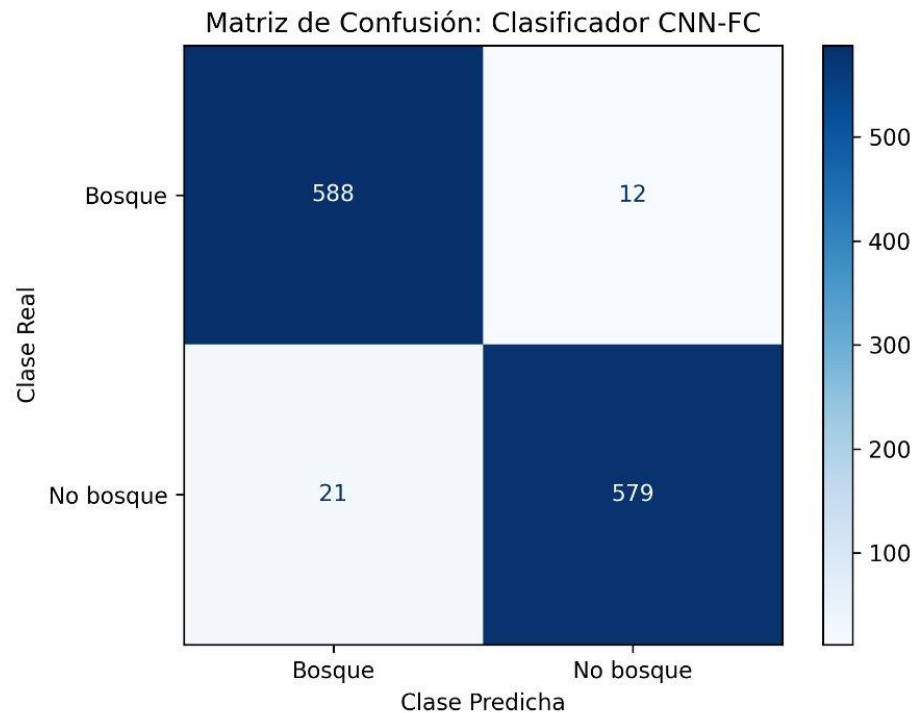
"Bosque"



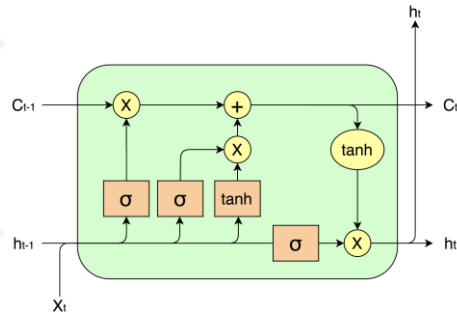
"No bosque"



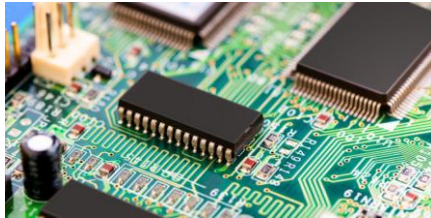
Clasificación de imágenes con SNN



Líneas futuras



- **Beamforming Digital mediante LSTM**



- **Aprendizaje en Hardware**



- **Implementación en tarjeta Versal**

Outcome

- *Enhancing efficiency in spaceborn phased array systems: MVDR algorithm and FPGA integration*, Digital Signal Processing (publicado)
- *FPGA-Based Adaptive Beamforming: Comparing Mvdr and Neural Networks for Spaceborne Systems*, Digital Signal Processing (en revisión)
- *A New Highly Efficient and FPGA-Oriented STDP-Based Spiking Neural Network for Topological Clustering* (en desarrollo)
- *Spiking Neural Networks for Earth Observation: A Hybrid Approach for Deforestation Detection* (en desarrollo)



Eduardo.Ortega@uah.es

Ruben.marco@uah.es

P.lopezgonzalez@uah.es

Agustin.martinez@uah.es

¡Muchas gracias!