Journal of Superintelligence and Computing

CoSci Press

Article

SpikeYOLO-C2TCA: An Improved Spiking Convolutional Neural Network for Remote Sensing Image Object Detection

Xianyi Wu [1,*], Guoyan Wang [2], Sergey Ablameyko [1,3]

[1] Belarusian State University, Minsk 220030, Belarus

[2] National Key Laboratory on ATR, National University of Defense Technology, Changsha 410073, China

[3] United Institute of Informatics Problems, National Academy of Sciences of Belarus, Minsk 220070, Belarus

* Corresponding author: tigerv5872@gmail.com

Article History: Received 9 June 2026; Revised 31 July 2026; Accepted 10 September 2026; Published 24 September 2026

Abstract: Despite their well-known energy advantages, spiking neural networks (SNNs) still fall short of conventional artificial neural networks in detection accuracy for on-orbit remote sensing, limiting their use on power-constrained satellites. To address this gap, we present SpikeYOLO-C2TCA, an improved SNN detector built upon the SpikeYOLO backbone. The core of our method is a new attention module, SpikeTCA, which operates directly on membrane potentials. It combines three complementary descriptors—direction-aware pooling, multi-scale depthwise convolution, and temporal convolution—and fuses them through cross-temporal self-attention and a lightweight channel MLP to enhance feature discriminability and suppress background clutter. In addition, we introduce a SpikeC2fCIB module that employs a five-layer depthwise-separable bottleneck following a split–process–merge structure, leading to better feature reuse, gradient flow, and computational efficiency. On the RSOD dataset, SpikeYOLO-C2TCA reaches an mAP50 of 96.7% and an mAP50-95 of 63.7%, exceeding the SpikeYOLO baseline by 6.6 and 1.2 percentage points, respectively. Overall, SpikeYOLO-C2TCA delivers better detection accuracy than the baseline while consuming less energy, offering a practical lightweight solution for edge platforms such as on-orbit satellites and long-endurance unmanned aerial vehicles.

Keywords: remote sensing image; object detection; spiking neural network; SpikeYOLO

1. Introduction

Satellite Earth observation has entered an era characterized by multi-satellite constellations, high-resolution imaging, real-time response, and global coverage. On-board data processing has emerged as a primary approach to enhancing the real-time capabilities of remote sensing data handling. However, challenges such as limited computational resources, stringent real-time requirements, and the unique space environment pose significant obstacles to on-orbit intelligence [1]. Object detection in high-resolution remote sensing imagery has become a crucial direction for spatial information acquisition, with applications spanning traffic planning, precision agriculture, military reconnaissance, and ocean management [2].

In recent years, deep learning-based object detectors, particularly Convolutional Neural Networks (CNNs) such as the R-CNN series [3–5] and the YOLO (You Only Look Once) family [6–9], have achieved remarkable accuracy. Nevertheless, their reliance on dense Multiply-Accumulate (MAC) operations results in high computational costs and energy consumption, severely restricting deployability on resource-constrained spaceborne or edge platforms.

Spiking Neural Networks (SNNs) offer a compelling alternative due to their event-driven computation: neurons transmit information via sparse binary spikes and consume energy only upon activation, drastically reducing power consumption [10]. Among recent SNN-based detectors, SpikeYOLO [11] stands out as a pioneering work that successfully integrates the YOLO architecture into the spiking paradigm while preserving the inherent energy efficiency of SNNs. It achieves a favourable balance between detection performance and computational cost, making it a strong candidate for on-orbit and edge applications.

However, when directly applied to remote sensing imagery, SpikeYOLO still exhibits a notable accuracy gap compared to mainstream Artificial Neural Network (ANN) detectors. This is primarily due to two challenging characteristics of remote sensing data. First, severe scale variation: objects such as aircraft, overpasses, and oil tanks can occupy vastly different pixel areas within the same scene, making it difficult for the baseline SNN backbone to capture multi-scale contextual information effectively. Second, complex background clutter: remote sensing images often contain abundant irrelevant textures, e.g., roads, vegetation, and water surfaces, that interfere with object localization, leading to false positives and missed detections. These issues constrain SpikeYOLO's detection precision and recall in real-world remote sensing scenarios.

To address these limitations, we propose SpikeYOLO-C2TCA, an enhanced SNN detector that introduces two spiking-compatible modules built upon the SpikeYOLO architecture:

SpikeC2fCIB, a spiking adaptation of the C2fCIB structure from YOLOv10 [12]. It employs a split-process-merge paradigm with SpikeCIB bottleneck blocks that incorporate depthwise separable convolutions and residual connections. This design enriches multi-scale feature representations and improves the model's ability to handle objects of varying sizes, effectively mitigating the scale-variation challenge in remote sensing.

SpikeTCA, a membrane-potential-level attention mechanism inspired by efficient multi-scale attention principles [13]. Unlike conventional spatial self-attention that relies on expensive matrix multiplications over spatial dimensions, SpikeTCA extracts complementary descriptors through three parallel branches, direction-aware pooling, multi-scale depthwise convolution, and temporal convolution, and fuses them via cross-temporal self-attention and a lightweight channel MLP. It generates adaptive channel weights that suppress background noise and highlight salient objects, while operating entirely on membrane potentials to preserve sub-threshold information. A bounded scaling mechanism ensures stable training by maintaining approximate identity mapping at initialization.

By integrating these two complementary modules into the deep feature pathway of SpikeYOLO, our method significantly enhances detection accuracy while maintaining the low-energy advantage of SNNs.

2. SpikeYOLO-C2TCA

The overall architecture of the proposed SpikeYOLO-C2TCA is built upon the SpikeYOLO framework and incorporates two spiking-compatible modules, SpikeC2fCIB and SpikeTCA, cascaded in the deep feature path of the detection head to form an efficient spiking feature enhancement unit.

Before detailing these modules, we first define the fundamental components and notations used throughout this section. Our input denotes a five-dimensional spiking data tensor, where T is the number of time steps, B is the batch size, and C, H, and W represent the channel, height, and width, respectively. SpikeConv operation is defined as a sequence of standard convolution, batch normalization, and Leaky Integrate-and-Fire (LIF) activation, ensuring that intermediate features remain in spike form.

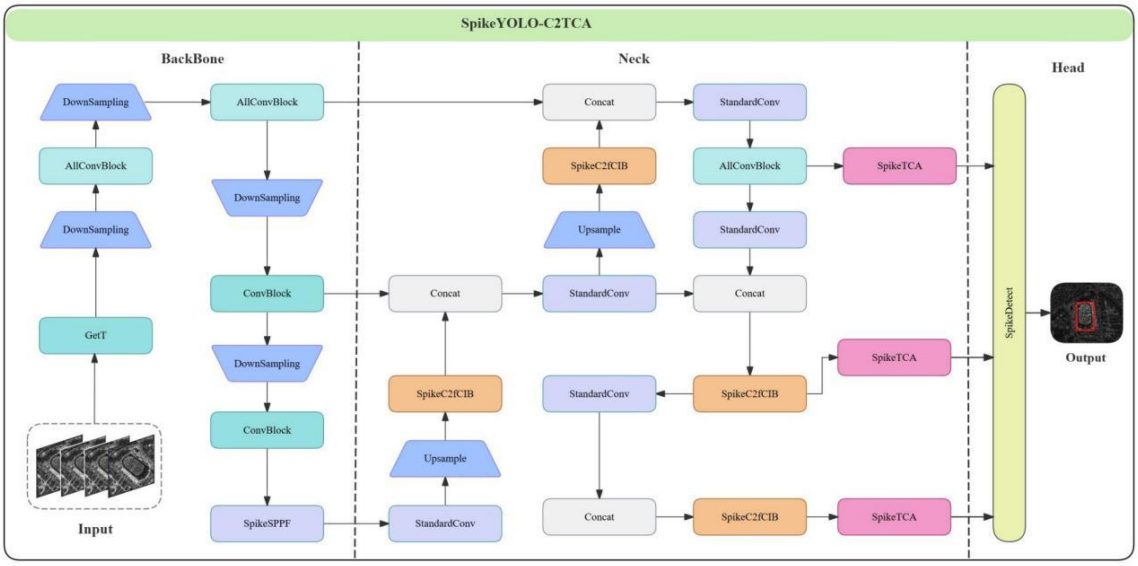


Figure 1. Schematic diagram of SpikeYOLO-C2TCA structure.

As illustrated in Figure 1, the proposed architecture integrates SpikeC2fCIB modules into all three scale levels of the detection head, P3 (80×80), P4 (40×40), and P5 (20×20), to enrich multi-scale feature representations throughout the feature pyramid. Specifically, SpikeC2fCIB is inserted after each upsampling stage and at the output layers of the head: following the P4 upsampling path, the P3 upsampling path, and at the final output branches for P3, P4, and P5. This multi-path design strengthens the detector's response to objects of varying sizes across all scales. Then, the SpikeTCA module is applied independently to each of the three detection branches (P3, P4, and P5), generating channel attention weights through grouping, coordinate pooling, and bidirectional interactions. This effectively suppresses background clutter and highlights salient target regions at every scale. The combination of these two modules enables SpikeYOLO-C2TCA to achieve markedly improved detection precision and recall while preserving the energy-efficient spike-driven nature of SNNs.

2.1. SpikeCIB

SpikeCIB is the fundamental bottleneck module used to construct SpikeC2fCIB, specifically designed for spiking neural networks. As illustrated in Figure 2, its core structure consists of five layers, depthwise separable convolutions alternately stacked with spiking neurons in a DWConv–PWConv–DWConv–PWConv–DWConv pattern, where DWConv denotes Depthwise Convolution and PWConv denotes Pointwise Convolution, with an intermediate channel expansion to 2c′, incorporating residual connections to efficiently extract local features while maintaining spiking-domain characteristics.

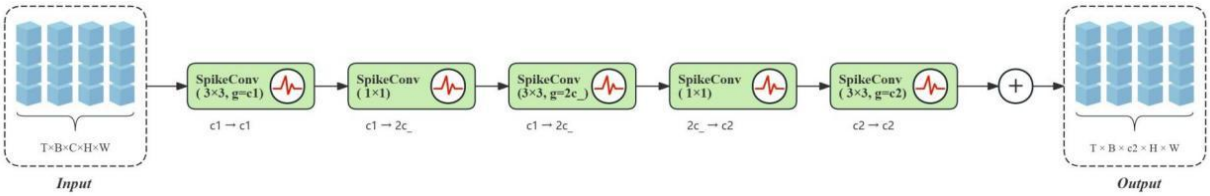


Figure 2. Schematic diagram of SpikeCIB structure.

In terms of implementation, the input first passes through a depthwise convolution (3×3, groups=c_1) followed by a spiking neuron (SpikeConv) to capture spatial information. Subsequently, a 1×1 pointwise convolution expands the channel dimension to 2c′, where $c' = c_2 \cdot e$ is the hidden channel width. Another depthwise convolution (3×3, groups=2c′) then processes the expanded features, followed by a 1×1 pointwise convolution that compresses the channels to the target output dimension c_2. Finally, an additional depthwise convolution (3×3, groups=c_2) performs spatial feature aggregation at the output stage. LIF neurons are integrated inside each SpikeConv layer to ensure that intermediate features remain in spike form. When the number of input channels equals the number of output channels and shortcut is set to True, the module adds the input to the output via an identity connection, forming a residual structure that effectively alleviates gradient vanishing and promotes feature reuse. By replacing standard convolutions with lightweight depthwise separable designs, this five-layer bottleneck expands the receptive field and enhances feature representation capability with limited parameters, serving as the core unit enabling multi-path gradient propagation in SpikeC2fCIB.

2.2. SpikeC2fCIB

SpikeC2fCIB is a spiking adaptation of the C2fCIB structure from YOLOv10. It first replaces all standard convolutions with their spiking-compatible counterparts (SpikeConv), and then inserts spiking neurons at critical locations. In our architecture, SpikeC2fCIB modules are deployed at multiple scales within the detection head, after upsampling stages and at the output layers for P3, P4, and P5, to enrich feature representations across the entire feature pyramid.

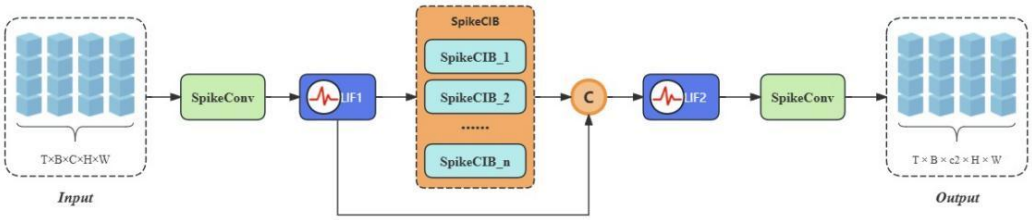


Figure 3. Schematic diagram of SpikeC2fCIB structure.

As illustrated in Figure 3, this module follows a split-process-merge paradigm, enhancing feature reuse and representation while maintaining computational efficiency, and is specifically tailored to the temporal dynamics of spiking neural networks (SNNs). The implementation process consists of three main steps:

First, the input tensor is passed through a 1×1 spiking convolution (SpikeConv), which maps the input channel dimension from C to 2c, where $c = c_2 \cdot e$, with c_2 being the target output channel dimension and e the expansion factor. After activation by an LIF neuron, it is evenly split along the channel dimension into two branches: one branch is preserved directly, while the other sequentially passes through multiple SpikeCIB bottleneck modules. All branches are concatenated along the channel dimension and subsequently compressed to the target channel dimension via a 1×1 spiking convolution.

Subsequently, branch B sequentially passes through n SpikeCIB modules, each maintaining the same dimensions, producing an output sequence $B^1, B^2, \ldots, B^n$.

Finally, all branches ($A, B, B^1, B^2, \ldots, B^n$) are concatenated along the channel dimension. After Leaky Integrate-and-Fire (LIF) activation, the concatenated features are compressed to the target channel dimension c_2 via a 1×1 spiking convolution, yielding the output.

2.3. SpikeTCA

Existing channel-attention mechanisms for spiking neural networks either ignore temporal correlations or operate on binary spikes after thresholding, discarding sub-threshold information in membrane potentials. SpikeTCA addresses both limitations by attending directly to the five-dimensional membrane-potential tensor $U \in \mathbb{R}^{T\times B\times C\times H\times W}$ and producing a same-shape attended potential U_{att}.

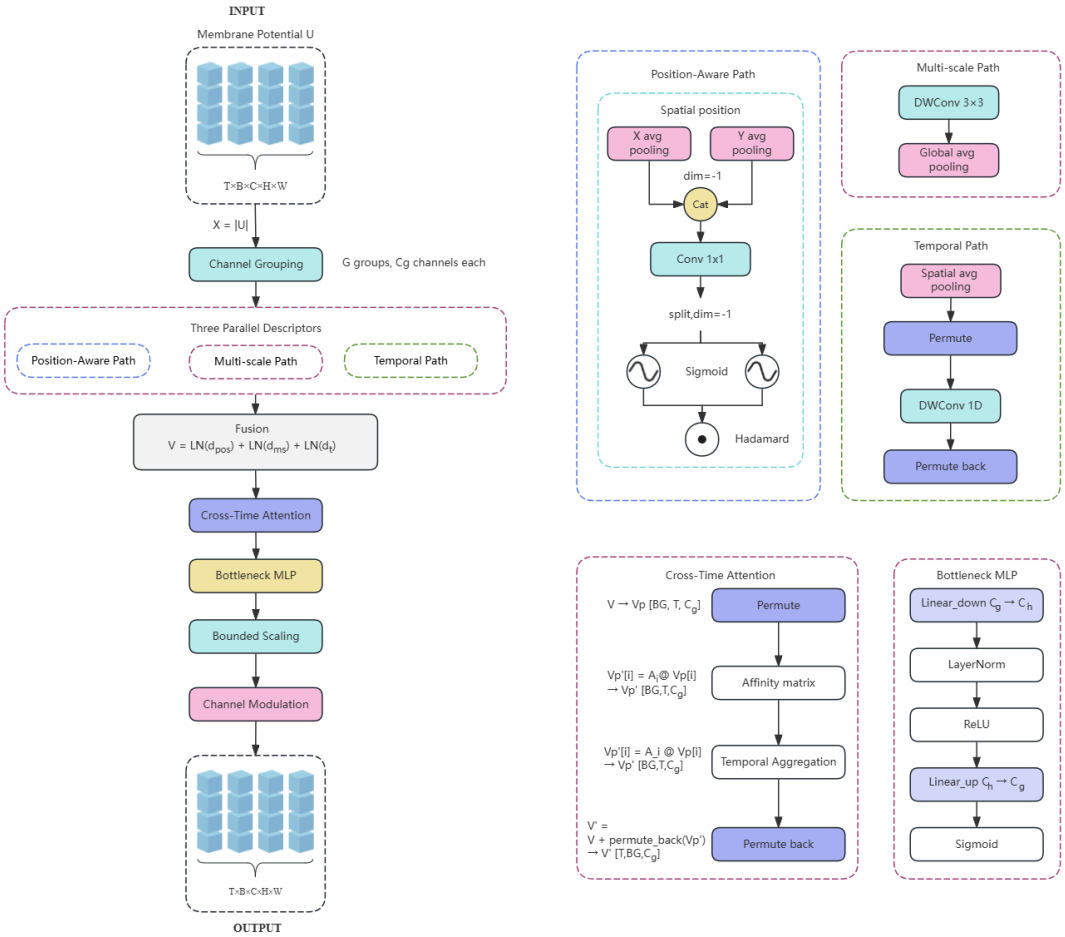


Figure 4. Schematic diagram of SpikeTCA structure.

The module first establishes a non-negative feature basis. Because membrane potentials can be negative, we take the element-wise absolute value, yielding the non-negative feature basis X = |U|, which preserves sub-threshold magnitudes that would be lost by thresholding to spikes. Channels are then partitioned into G = 8 groups and reshaped into $X_g \in \mathbb{R}T\times(BG)\times Cg\times H\times W$ with Cg = C/G, where BG = B·G (Batch times Group).

Descriptor extraction. Three complementary descriptors are extracted in parallel from X_g. The multi-scale descriptor applies a 3×3 depthwise convolution followed by global average pooling. The temporal descriptor spatially pools X_g to $\mathbb{R}^{T\times BG\times Cg}$, permutes time to the last dimension, and applies a depthwise 1-D convolution with kernel size 3 to capture temporal evolution. The direction-aware descriptor, which encodes spatial position without an explicit 2-D attention map, pools X_g along the horizontal and vertical axes, projects the concatenated statistics through a shared 1×1 convolution, splits the result into a_h and a_w, and aggregates them as

$$d_{pos} = \left(\frac{1}{H}\Sigma_h\sigma(a_h)\right) \odot \left(\frac{1}{W}\Sigma_w\sigma(a_w)\right) \quad (1)$$

Fusion and cross-time attention. The three descriptors are layer-normalized, summed into V, and refined by lightweight cross-time self-attention. Permuting V to $V_p \in \mathbb{R}^{BG\times T\times Cg}$, the attended descriptor with residual connection is

$$V' = V + Permute^{-1}\left(softmax\left(V_p V^{pT}/\sqrt{C_g}\right)V_p\right) \quad (2)$$

where V' denotes the descriptor after cross-time self-attention, and the softmax is applied independently per batch-group index. Because T ≤ 8 in typical SNN configurations, the T× T matrix product adds negligible cost.

Channel-weight generation and modulation. Using σ(·) for the Sigmoid function and LN for Layer Normalization, a bottleneck MLP with hidden dimension C_h= max(C_g/r, 4) (r = 16) maps V' to raw channel weights. A learnable scalar α (initialized to 0) provides soft gating, yielding the final modulation weights

$$W = 1 + tanh(\alpha)\cdot 2\sigma\left(Linear^{\uparrow}\left(ReLU\left(LN\left(Linear^{\downarrow}(V')\right)\right)\right)\right) \quad (3)$$

where *Linear*↓ and *Linear*↑ denote down-projection and up-projection linear layers, respectively, with Linear↓: $\mathbb{R}^{Cg}\rightarrow \mathbb{R}^{Ch}$and Linear↑: $\mathbb{R}^{Ch} \rightarrow \mathbb{R}^{Cg}$. At initialization α = 0, so W ≡ 1 and the module acts as an identity mapping. The modulated membrane potential is:

$$U_{att} = U \odot W \quad (4)$$

Because U_{att} retains the membrane-potential format, it can be fed directly to the next spiking layer for thresholding and gradient propagation.

3. Experiments and Results Analysis

3.1. Datasets

In this paper, a series of comparative and ablation experiments were conducted based on the open-source remote sensing image object detection dataset RSOD [14,15]. The RSOD dataset, released by Wuhan University in 2015, is a specialized dataset for remote sensing object detection. It includes: 4,993 aircraft instances across 446 images. 191 playground instances across 189 images. 180 overpass instances across 176 images. 1,586 oil tank instances across 165 images.

3.2. Experimental Setup and Evaluation Metrics

All experiments in this study were conducted under a unified software and hardware environment to ensure the reliability of the results and the accuracy of the data. The detailed configuration parameters of the experimental environment are presented in Table 1. The number of training epochs was set to 300. For any parameters not explicitly mentioned in this paper, the default values from SpikeYOLO were adopted.

Table 1. Experiment configuration table.

Environment		Parameter	
Operating System	ubuntu22. 04	Learning Rate	0. 01
GPU	RTX 4090(24GB) * 1	Iterations	300
Memory	120G	Batch size	40
Python	Python 3. 10(ubuntu22. 04)	Workers	8
Framework	PyTorch 2. 1. 0	Image Input Size	640×640
Environment	CUDA 12. 1	optimizer	auto

To comprehensively evaluate the performance of the proposed model, we adopted Precision, Recall, Average Precision (AP), and mean Average Precision (mAP) as the primary evaluation metrics. The calculation methods for each indicator are shown in Table 2.

Table 2. Evaluation indicators.

Symbol	Evaluation indicators
AP	Average precision
P	Precision
R	Recall
IoU	Intersection over Union
mAP50	mean Average Precision (IoU=0. 5)
mAP 50-95	mean Average precision (IoU=0. 5-0. 95)

We evaluated the proposed method on the remote sensing dataset RSOD. For each model, we report the mean average precision at IoU=0. 5 (mAP@50), the average mAP over IoUs from 0. 5 to 0. 95 (mAP@50:95), and the energy consumption. Specifically, the power consumption formulas for ANN and SNN are as follows:

$$E_{ANN} = O^2\times C_{in}\times C_{out}\times K^2\times E_{MAC} \quad (5)$$

$$E_{SNN} = (T\times D)\times f_r\times O^2\times C_{in}\times C_{out}\times K^2\times E_{AC} \quad (6)$$

where O is the feature output size, Cin and Cout denote the number of input channel and output channel, k is the kernel size, fr denotes the average spike firing rate, T is the timestep, D is the upper limit of integer activation during training. We follow the most commonly used energy consumption evaluation method in the SNN field [16]. All operations assume a 32-bit floating-point implementation on 45nm technology, where EMAC = 4.6 pJ and EAC = 0.9 pJ [17].

3.3. Comparative Experiments

We compared SpikeYOLO-C2TCA against YOLOv5n (nano), YOLOv8n (nano), YOLOv8m (medium), and the original SpikeYOLO on the RSOD dataset. Results are summarized in Table 3.

Table 3. Comparison results of RSOD remote sensing datasets.

Model	P (%)	R (%)	mAP50 (%)	mAP50-95 (%)	Parameters (M)	GFLOPs	Power (mj)
YOLOv5n	95.4	83.2	91.5	64.4	2.50M	7.1	16.33
YOLOv8n	97.5	89.4	95.4	64.6	3.01M	8.1	18.63
YOLOv8m	96.5	87.5	93.8	67.9	25.84M	78.7	181.01
SpikeYOLO	89.6	84.6	90.1	62.5	13.25M	78.1	13.21
Ours	96.9	92.3	96.7	63.7	9.82M	51.6	8.65

Table 4 reports energy consumption across time-step configurations. The 4×4 setting gives the best accuracy-energy trade-off at 96.7% mAP50 and 8.65 mJ. The 4×8 setting reaches 66.2% mean Average Precision at IoU=0.5:0.95 (mAP50-95) at 11.9 mJ. Our method exceeds SpikeYOLO in accuracy and energy at every configuration: at 4×8 it leads by 2.3 percentage points in mAP50 while using 26.4% less energy, and at 4×4 the margin widens to 6.6 percentage points with energy cut by 34.5%. The reduction comes from SpikeC2fCIB's depthwise separable convolutions, which lower active synaptic operations. Even at the extreme 1×1 setting, our model retains 72.5% mAP50 while consuming 5.2 mJ, 33.8% less energy than the baseline at the same configuration. EMS-YOLO, an ANN-to-SNN converted detector, consumes 85.6 mJ at T=5, nearly ten times our 4×4 setting, while reaching only 78.71% mAP50. Even at T=1 it uses 6.4 mJ with accuracy at 69.06% mAP50. The steep energy scaling reflects a limitation of conversion-based methods: they inherit dense ANN architectures that cannot exploit temporal sparsity. Natively trained spike-driven detectors avoid this penalty.

Table 4. Power consumption comparison across time-step configurations.

Model	T×D	mAP50 (%)	mAP50-95 (%)	Power (mJ)
Ours	4×8	95.5	66.2	11.9
	4×4	96.7	63.7	8.65
	2×2	87.2	59.9	7.1
	1×1	72.5	45.3	5.2
SpikeYOLO	4×8	93.2	65.5	16.16
	4×4	90.1	62.5	13.21
	2×2	90	60.8	10.76
	1×1	76.3	49.3	7.86
EMS-YOLO	5	78.71	40.96	85.6
	4	77.83	39.04	77.1
	2	73.84	37.78	28.6
	1	69.06	32.1	6.4

3.4. Model Comparison Experiments

We compared SpikeYOLO and our method category by category on RSOD. Table 5 lists the results for aircraft, oil tank, overpass, and playground. All experiments used identical training and inference conditions.

We also tested generalization on DIOR, a large-scale benchmark with 20 categories, 23,463 images, and 192,472 object instances. Training ran for 300 epochs under the same hyperparameters. Results are in Table 6.

Table 5. Comparison results of all categories in the RSOD dataset.

	SpikeYOLO				Ours			
	P (%)	R (%)	mAP50 (%)	mAP50-95 (%)	P (%)	R (%)	mAP50 (%)	mAP50-95 (%)

all	89.6	84.6	90.1	62.5	96.9	92.3	96.7	63.7
aircraft	89.1	92.7	94.9	63.6	92.1	92.7	96.2	63.9
oil tank	98.1	92.9	96.2	77.2	97.9	94.2	97.4	77.0
overpass	82.3	52.6	69.7	27.1	100.0	87.4	95.1	32.1
playground	88.8	100.0	99.5	81.9	97.6	94.7	98.3	81.6

Table 6. Comparison results on the DIOR dataset.

	SpikeYOLO		Ours	
all	86	63.3	87.2	64.7
airplane	96.1	72.5	96.5	73.7
airport	93.6	71.5	95.6	74.4
baseball field	94.5	81.5	95.4	82.8
basketball court	89.3	79.3	90	80.4
bridge	61.8	38.4	64.1	39.4
chimney	91.2	79.4	92.1	81.3
dam	84.7	52.1	87.1	54.1
Expressway-Service-	95.8	74.1	97.3	76.5
Expressway-toll-sta-	85.1	66.1	86	67.8
golf field	89	71.6	88.9	71.7
ground track field	89.9	73.1	90.7	74.5
harbor	71.3	51.7	73	52.4
overpass	70.3	47.8	71.6	47.8
ship	93	57	93.3	57.6
stadium	96.4	80	96.4	82.6
storage tank	87.8	56.4	88.4	56.9
tennis court	95.7	83.8	95.8	84.5
train station	76.5	43.4	81.8	46.8
vehicle	65.9	37.5	67.3	38.7
windmill	91.6	49.6	92.6	50.6

On the DIOR dataset, our method attains an mAP50 of 87.2% and an mAP50-95 of 64.7%, representing improvements of 1.2 and 1.4 percentage points over the SpikeYOLO baseline (86.0% / 63.3%). These gains are smaller than those observed on RSOD, which is unsurprising given DIOR's wider variations in object scale, orientation, and background complexity [18]. The consistent improvement across both datasets indicates that the proposed modules generalize effectively to larger and more challenging remote sensing scenes. Category-level analysis reveals that the largest gains occur in structurally complex or clutter-prone classes. trainstation improves by 5.3 percentage points in mAP50, highlighting SpikeTCA's ability to suppress cluttered backgrounds around dense urban structures. Dam and bridge rise by 2.4 and 2.3 percentage points respectively, consistent with the multi-scale feature fusion provided by SpikeC2fCIB. Airport also gains 2.0 percentage points, reflecting improved handling of large-scale infrastructure with heterogeneous textures. golffield shows a minor decrease of 0.1%, attributable to its small sample size and high intra-class similarity, which limits the benefit of attention modulation. Stadium remains unchanged at 96.4% mAP50, suggesting this class was already near saturation for both models. Overall, 18 of the 20 categories achieve improved or equal mAP50.

3.5. Ablation Experiment and Random Seed Experiment

We conducted ablation experiments on the RSOD dataset using SpikeYOLO as the baseline to isolate the contribution of each proposed module. Components were added gradually; results are in Table 7.

Table 7 shows that the two modules are complementary. Adding SpikeTCA alone raises mAP50 from 90.1% to 95.6% and mean Average Precision at IoU=0.5:0.95 (mAP50-95) from 62.5% to 64.5%, with only a slight increase in parameters. Adding SpikeC2fCIB alone cuts parameters by 25.9% and GFLOPs by 34.1%, while still lifting mAP50 to 93.7%. Combining both modules yields the best result: 96.7% mAP50 at 63.7% mean Average Precision at IoU=0.5:0.95 (mAP50-95), with parameters and GFLOPs close to the SpikeC2fCIB-only level. SpikeTCA improves detection accuracy, and SpikeC2fCIB reduces cost.

Table 7. Ablation experiment results of RSOD dataset.

SpikeYOLO	SpikeTCA	SpikeC2fCIB	P (%)	R (%)	mAP50 (%)	mAP50-95 (%)
√			89.6	84.6	90.1	62.5
√	√		96.0	91.1	95.6	64.5
√		√	94.7	91.3	93.7	63.5
√	√	√	96.9	92.3	96.7	63.7

Table 8. Random seed robustness experiment on RSOD dataset.

Model	Metric	Seed									mean ± std
		0	43	101	256	456	789	2026	2111	2037	
SpikeYOLO	mAP50	90.1	94.8	94.7	91.6	95	92.3	94.5	93.2	94.7	93.43±1.75
Ours	mAP50	96.7	93.8	94.2	96.2	96.6	95.2	94.2	95.2	95.9	95.33±1.09

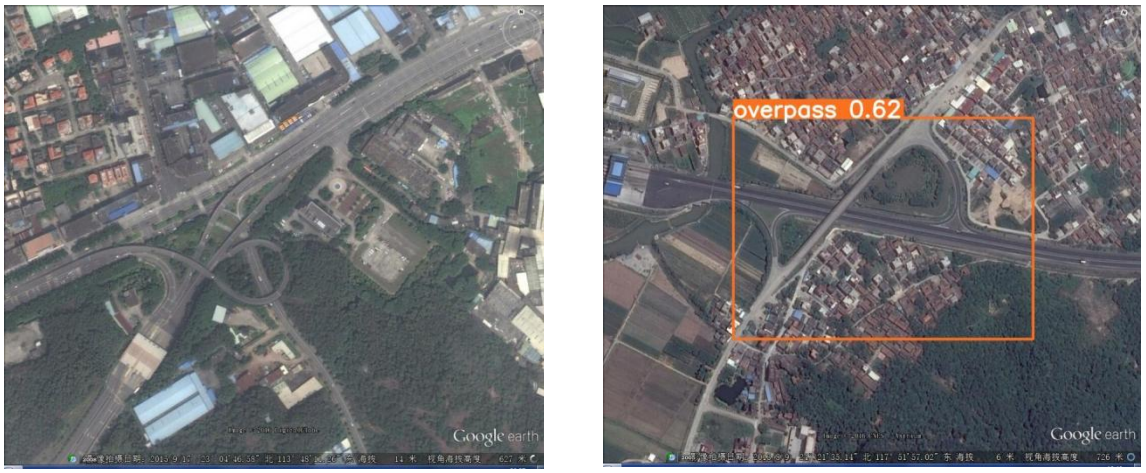
To assess statistical robustness, we trained both models with nine independent random seeds on the RSOD dataset. Results are summarized in Table 8. SpikeYOLO achieves 93.43 ± 1.75% mAP50, whereas SpikeYOLO-C2TCA attains 95.33 ± 1.09% mAP50. The mean improvement is 1.9 percentage points, and our method shows notably lower variance, indicating superior training stability. While the gap reported in the main comparison (6.6 pp at seed 0) reflects a specific initialization, the consistent mean advantage across seeds confirms the reliability of the proposed modules.

As shown in Figure 5, we tested the SpikeYOLO algorithm and the proposed SpikeYOLO-C2TCA algorithm. In the aircraft image detection shown in Figure 5(a), SpikeYOLO produced an obvious false positive on the left side. In Figure 5(b), which depicts an image with a complex background, SpikeYOLO failed to detect the overpass. In Figure 5(c), where the oil tanks have a similar color to the background, SpikeYOLO was unable to detect the oil tanks in the bottom row.

3.6. Visual Analysis



a) Aircraft detection result comparison



b) Overpass detection result comparison



c) Oiltank test results comparison

SpikeYOLO SpikeYOLO-C2TCA

Figure 5. Comparison of detection methods between SpikeYOLO and SpikeYOLO-C2TCA.

4. Conclusion

We presented SpikeYOLO-C2TCA, a spiking neural network for remote sensing object detection that improves accuracy while reducing computational cost [19]. The model introduces two modules: SpikeC2fCIB, which compresses parameters and operations by replacing standard convolutions with depthwise separable spiking convolutions in a C2f (Cross-stage Partial with 2 convolutions) bottleneck; and SpikeTCA, a temporal-channel attention mechanism that operates on membrane potentials to capture spatial, multi-scale, and temporal cues.

On the RSOD dataset, SpikeYOLO-C2TCA achieves 96.7% mAP50 and 63.7% mean Average Precision at IoU=0.5:0.95 (mAP50-95), surpassing the SpikeYOLO baseline by 6.6 and 1.2 percentage points respectively, with 25.9% fewer parameters and 33.9% lower computational cost. Energy consumption per inference is 8.65 mJ, 34.5% lower than the baseline. Ablations confirm that the two modules play complementary roles: SpikeTCA primarily improves localization accuracy, while SpikeC2fCIB drives efficiency gains.

These improvements come with a boundary. The current architecture fixes the time step and channel group count, which may not be optimal for all scene types. Future work will explore adaptive time-step mechanisms and hardware-aware quantization for deployment on neuromorphic chips.

Future work will proceed along two lines: first, exploring adaptive time-step mechanisms to find a more flexible balance between accuracy and latency. Second, investigating hardware-aware quantization and pruning strategies for deployment on neuromorphic chips.

Author Contributions: Conceptualization, methodology, writing—original draft preparation, investigation, X.W.; writing—review & editing, formal analysis, G.W.; supervision, project administration, S.A. All the authors have read and agreed to the published version of the manuscript.

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Ethical Approval: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: None.

Conflicts of Interest: The authors declare no conflicts of interest.

References

[1] Yang, N.; Zhu, Q.; Wu, P.; Wang, Y.; An, J.; Sun, F. Research on the progress of on-orbit processing of satellite remote sensing images. *Space Electron. Technol.* **2023**, *20*, 1–8.

[2] Zhang, S.; Li, S.; Wei, G.; Zhang, X.; Gao, J. Refined multi-scale feature-oriented object detection of remote sensing images. *Natl. Remote Sens. Bull.* **2022**, *26*, 2616–2628.

[3] Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.

[4] Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.

[5] Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149.

[6] Khanam, R.; Hussain, M. What is YOLOv5: A deep look into the internal features of the popular object detector. *arXiv* **2024**, arXiv:2407.20892 (preprint).

[7] Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; et al. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv* **2022**, arXiv:2209.02976 (preprint).

[8] Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.

[9] Zhou, C.; Zhu, Y.; Zhang, J.; Ding, Z.; Jiang, W.; Zhang, K. The tea buds detection and yield estimation method based on optimized YOLOv8. *Sci. Hortic.* **2024**, *338*, 113730.

[10] Liu, H.; Chai, H.; Sun, Q.; Yun, X.; Li, X. Research status and application progress of spiking neural networks. *Strateg. Study CAE* **2023**, *25*, 61–79.

[11] Luo, X.; Yao, M.; Chou, Y.; Xu, B.; Li, G. Integer-valued training and spike-driven inference spiking neural network for high-performance and energy-efficient object detection. *arXiv* **2025**, arXiv:2407.20708 (preprint).

[12] Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; et al. YOLOv10: Real-time end-to-end object detection. *arXiv* **2024**, arXiv:2405.14458 (preprint).

[13] Ouyang, D.; He, S.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; et al. Efficient multi-scale attention module with cross-spatial learning. In Proceedings of the ICASSP 2023 – 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.

[14] Long, Y.; Gong, Y.; Xiao, Z.; Liu, Q. Accurate object localization in remote sensing images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2486–2498.

[15] Xiao, Z.; Liu, Q.; Tang, G.; Zhai, X. Elliptic Fourier transformation-based histograms of oriented gradients for rotationally invariant object detection in remote-sensing images. *Int. J. Remote Sens.* **2015**, *36*, 618–644.

[16] Panda, P.; Aketi, S.A.; Roy, K. Toward scalable, efficient, and accurate deep spiking neural networks with backward residual connections, stochastic softmax, and hybridization. *Front. Neurosci.* **2020**, *14*, 653.

[17] Horowitz, M. 1.1 Computing's energy problem (and what we can do about it). In Proceedings of the 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), San Francisco, CA, USA, 9–13 February 2014; pp. 10–14.

[18] Mou, D. A Comprehensive Survey of Deep Learning–Based Object Tracking for Augmented Reality in Complex Real-World Scenes. *Appl. Artif. Intell. Res.* **2026**, *2*(1).

[19] Li, H.; Zhao, X. Research on Auxiliary Diagnosis of Thoracic Fractures Based on Bidirectional Feature Pyramid Network. *Int. J. Adv. AI Appl.* **2025**, *1*, 23–32.

Copyright: © 2026 by the author(s). *Journal of Superintelligence and Computing* supported by Wuchang University of Technology and published by CoSci Press. This is an open access publication under the Creative Commons Attribution 4.0 International License (CC BY 4.0).