Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

S3Det: A fast object detector for remote sensing images based on analog-spiking neural network conversion

LiChen[†1], FanZhang[†‡1], Guangwei Xie [†2], Yanzhao Gao[†1], Xiaofeng Qi [†1], Mingqian Sun [†‡3]

[1] *National Digital Switching System Engineering & Technological R&D Center, Zhengzhou, Henan 450003, China.*

[2] *School of Computer Science, Fudan University, Shanghai 201203, China.*

[2] *College of Cyber Science and Engineering, Southeast University, Nanjing 211189, China.*

[†]E-mail: zhangfanryan@163.com

Received July 14, 2024; Revision accepted Oct. 11, 2024; Crosschecked

Abstract: Artificial neural networks (ANNs) have made great strides in the field of remote sensing image object detection. However, low detection efficiency and high power consumption have always been a significant bottleneck in remote sensing. Spiking neural networks (SNNs) process information in the form of sparse spikes, creating the advantage of high energy efficiency for computer vision tasks. However, most work has focused on simple classification tasks, and only a few researchers have applied SNNs to object detection in natural images. In this study, we consider the parsimonious nature of biological brains and propose a fast ANN-to-SNN conversion method for remote sensing image detection. We establish a fast sparse model for pulse sequence perception based on group sparse features and conduct transform-domain sparse resampling of the original image to enable fast perception of image features and encoded pulse sequences. In addition, to meet accuracy requirements in relevant remote sensing scenarios, we analyze the transformation error theoretically and propose channel self-decaying weighted normalization to eliminate neuron overactivation. We propose S3Det, a remote sensing image object detection model. Our experiments, based on a large publicly available remote sensing dataset, show that S3Det achieves an accuracy performance similar to that of the ANN. Meanwhile, our transformed network is only 24% sparser than the benchmark and consumes only 1.46 J. On the simulation platform, our algorithm improves the integrated inference time 48 times more than the benchmark and consumes a fraction (1/122) of the original algorithm's power.

Key words: Remote sensing image; Object detection; Spiking neural networks (SNNs); Spiking sequence rapid sensing; Channel self-decay normalization

https:/https://doi.org/10.1631/FITEE.2400594 **CLC number:**

1 Introduction

Due to the rapid development of remote sensing technology in recent years, the automatic analysis of massive remote sensing images with all-round intelligent technology has become urgently needed by academia and industry. Object detection, a fundamental task in this domain, finds extensive application in essential areas such as meteorological analysis, surface surveying, and transportation planning (Chen et al., 2023).

‡ Corresponding author

* Project supported by the National Key Research and Development Program of China（No. 2022YFB4500900）

ORCID: Li CHEN, https://orcid.org/0009-0006-8206-5255; Fan ZHANG, https://orcid.org/0000-0001-7456-8377

Artificial neural networks (ANNs), represented by convolutional neural networks (CNNs) (Lecun et al., 1998) and Transformers (Vaswani, 2017), have developed rapidly over the past decade and continue to energize object detection tasks. However, ANNs often demand substantial computing resources, posing challenges for their deployment on resource-limited devices. In contrast, spiking neural networks (SNNs) (Maass, 1997) emulate the biological structure of the brain, encoding and transmitting information through 0-1 spikes, mimicking human biological systems. Because spiking neurons only trigger calculations in response to external stimuli, they have the huge advantages of low power consumption and fast reasoning. At present, SNNs have achieved excellent performance on a variety of hardware platforms, which also validates the feasibility of exploring high-performance and highly efficient brain-inspired intelligence through SNNs.

Despite binary spiking enabling the extreme energy efficiency of SNNs, the complex dynamics involved and the non-differentiable mathematical characteristics of spiking neurons result in a scarcity of training algorithms. Therefore, researchers try to find the balance of the scales in the conversion method. The ANN-to-SNN conversion (Kim et al., 2020; Li et al., 2022; Hu et al., 2023; Yao et al., 2023) has already yielded promising results on common object detection datasets, such as PASCAL VOC (Everingham et al., 2010) and COCO (Lin et al., 2014). However, as the number of network layers deepens, the accuracy error caused by the conversion method will be further amplified. While increasing the firing rate of discrete spikes can significantly mitigate these accuracy errors, it leads to a substantial increase in time steps, undermining the efficiency advantages of SNNs.

To fully harness SNNs' efficiency advantages in real-time remote sensing detection scenarios, our research draws on the principles of simplicity and sparsity inherent in biological information transmission. The simplicity of SNNs lies in their efficient coding, local computation, and event-driven nature. SNNs employ a simple encoding scheme to represent complex information, with neurons communicating primarily with their immediate neighbors, thereby reducing the energy consumption associated with long-distance signal transmission. The event-driven characteristic ensures that neurons generate spikes only when there is a change in the input, leading to a more energy-efficient operation. The sparsity of SNNs is manifested in three key respects: sparse activation, sparse connectivity, and sparse representation. Sparse activation refers to only a small fraction of neurons being active and generating spikes at any given time. This reduces the overall computational load and energy consumption. Sparse connectivity implies that the connections within the neural network are not fully dense; instead, each neuron typically connects to only a subset of other neurons. Sparse representation, on the other hand, involves using the minimum number of active neurons to encode information.

Guided by the aforementioned principles, we also incorporate the theory of compressive sensing from signal processing. Compressive sensing allows for the recovery of sparse signals from a small number of measurements. In the context of SNNs, this can be leveraged to reconstruct complete signals from sparse activation patterns, thereby reducing the number of required sensors and the amount of data. We developed a fast SNN model suitable for remote sensing applications, achieving a performance competitive with that of ANNs. The fundamental concept was to eliminate redundant computations and leverage the temporal and spatial features more effectively. In addition, we further analyzed the causes of errors in the conversion process and made corresponding improvements. The following summarizes the contributions we have made in this work and highlights the benefits of the lightweight SNN that has been proposed:

(1) Spiking sequence rapid sensing. We analyzed the sparse nature of spiking sequences in aerial images and modeled them using group sparse compressed sensing. Theoretically, we demonstrated that the proposed model effectively performs compressed sampling of the original spiking sequences. This is achieved through the minimization of a mixed norm model.

(2) Channel self-decaying weighted normalization. To address the issue of excessive activation of spiking neurons, we conducted an in-depth analysis of normalization errors during the conversion from ANN to SNN. Our findings indicated that the spikes of inactivated neurons (SIN) error was exacerbated as the number of layers and channels increased. This issue is particularly critical for neural networks that process remote sensing images and must be carefully considered. We recommended an exponentiated momentum decay scheme based on low-order statistics along the channel dimension, which offers a cost-effective solution to this problem.

(3) Depth model of remote sensing object detection. Our method developed SNN models for object detection in remote sensing images, achieving advanced performance and efficient detection on major publicly available datasets. To the best of our knowledge, this was the first attempt to apply SNNs to object detection tasks in the field of remote sensing.

2 Related Work

2.1 Remote sensing image object detection

Different from natural images, remote sensing images are captured from an overhead perspective and include a diverse array of objects such as vehicles, bridges, and ships. These objects are characterized by arbitrary orientations, relatively small and dense objects, and significant scale variations among objects. To address these, researchers have focused on three areas—feature refinement, anchor-free mechanisms, and the optimization of the loss function—to develop more accurate object detection algorithms for high-resolution remote sensing images.

In terms of feature refinement, it is necessary to effectively augment rotated image data to maintain rotational invariance. Cheng et al. (2016a) proposed a rotation-invariant convolutional neural network (RICNN) that enhances model generalization by utilizing sample rotations. Based on the original model, Cheng further incorporated a Fisher discriminant layer to enhance classification similarity (Cheng et al., 2016b), thereby improving classification performance. Additionally, various methods have been employed to refine features through data augmentation, including quad-patch augmentation (Gong et al., 2022), dual-dimensional feature enhancement using Multiscale Attention CycleGAN (Liu et al., 2021), and specifically synthetic mineral oversampling with mosaic and mixup (SSMup) (Chen et al., 2022), among others. Enhancing the semantic features of objects is also a common optimization method. Yang et al. (2021) proposed a progressive regression method called R3Det, which strengthens the accuracy by refining the center points of objects from coarse to fine granularity.

The two-stage detectors with the above-detailed features are all designed based on anchors. However, due to the enumeration and the parameter adjustment difficulty of anchors, researchers have gradually expanded the Anchor-free algorithm. In 2020, He et al. (2022) designed HRPNet, which converts the detection of oriented bounding boxes (OBBs) (as shown in Fig. 1b) into the regression of an angle and four radii in polar coordinates. This significantly reduced the computational complexity of the network model. In 2022, Zhang et al. (2023) used a coarse location module to quickly produce coarse oriented boxes, then employed a region-based convolutional neural network (R-CNN) to complete the detection and classification of objects. The above models eliminated the need for laborious manual anchor design, offering competitive detection speed. Inspired by the characteristics of densely distributed remote sensing objects, Xie et al. (2024a) proposed a method called the Objectness Activation Network (OAN), which predicts whether each image patch contains an object, thereby enhancing detection efficiency. Additionally, Xie et al. (2023) introduced the concept of collaborative learning to address the issues of non-universality in target features and the limitations of single-regression approaches. However, the CNN often suffered from significant feature loss during forward propagation, leading to missed detections and false positives. Consequently, some researchers have opted to optimize the reward and penalty mechanisms directly within the design of the loss function. In the field of remote sensing, the main challenges in designing loss functions are boundary discontinuity and inconsistency with the final detection metrics. To address the boundary discontinuity issue, Yang et al. (Yang et al. and Xue et al.) used the Gaussian Wasserstein Distance (GWD) to approximate the IOU rotational loss. Building on Yang's work, Huang et al. (2022) designed a novel General Gaussian Heatmap Label Assignment (GGHL), which not only creates 2D Gaussian heatmaps but also employs an anchor-free adaptive label assignment strategy to improve detection efficiency.

All the above methods seek data and network-level optimizations within the framework of ANNs, inherently limiting the upper bound of efficiency. Even when employing anchor-free single-stage detection algorithms, the resulting detection performance fails to meet the lightweight and low-latency requirements of the remote sensing domain. SNNs transmit information to downstream neurons using action potentials rather than the real values used in ANNs. Due to the sparsity of image inputs, SNNs exhibit significant efficiency advantages.

(a) (b)

Fig. 1 Two forms of remote sensing image detection box. The horizontal bounding box (HBB) shown in (a) contains excessive noise compared to the oriented bounding box (OBB) shown in (b)

3 S3Det

3.1 Network Architecture

Related work on the conversion of ANNs to SNNs and their proof of equivalence can be found in the Supplementary Materials. The overall network structure of S3Det is shown in Fig. 2. First, the image is pre-processed with conventional operations, including cropping, scaling, and rotation. The ANN algorithm used for conversion is R3Det. The conversion includes parsing the ANN network, parameter normalization, neuron conversion, and the post-process. In the converted S3Det, we designed a spiking sequence rapid sensing module (SSRS) and channel self-decaying weighted normalization (CSWN) based on low-order statistics.

Fig. 2 Architecture of the proposed S3Det detector. In the input image, the object to be detected is a tennis court

3.2 Spiking sequence rapid sensing module

In SNNs, the rate-based encoding method can be represented by the firing rate v as follows:

, (1)

where Nspike represents the number of spikes and T denotes the time window. For remote sensing images, the pixel values correspond to the number of spikes. However, the spike sequences obtained based on frequency are not sparse, necessitating the resampling of spike sequences Xorigin to X. If we use the L2 norm to measure the error in the resampling result, we can model the concept of compressed sensing as follows:

, (2)

where represents a compressed sampling operator in the sparse domain Rm; the sampling process for our sparse signal X can be expressed as:

, (3)

where represents the ith known reference sampling kernel function, n denotes the number of samples, and indicates the sampling error for the ith sample.

Because of the complex and redundant features of remote sensing images, to minimize information loss during compression, we need to identify a K-order restricted isometry constant (RIC) that satisfies the following equation, ensuring it is as small as possible to guarantee the robustness of information sampling:

(4)

However, typically solving for the K-order RIC constant of a matrix is an NP-hard problem. To address this, we draw inspiration from neuroscience, where the visual cortex of the brain can encode visual information with a minimal number of neurons. Thus, we consider using a group sparsity model that incorporates prior knowledge of the image to constrain the problem. The sparsity model can be defined as follows:

, (5)

where g represents the group structure.

In the initial spike sequence, there is a strong correlation between spike count and pixel intensity. As an inherent attribute of images, non-local similarity and group sparsity are widely utilized in image reconstruction. Therefore, we use a Gaussian mixture model (Komarek and Lesaffre, 2008) to capture the prior knowledge of the image. First, for a given remote sensing image patch, we identify the p-1 most similar image patches to form the image patch group Xp. Subtracting the group mean from these patches yields the corresponding residual group Xn. Assuming that the residuals of the same matrix belong to the same Gaussian component and are independent between matrices, the likelihood function of Xn can be expressed as follows:

, (6)

where is the Kth component of the Gaussian mixture, and is the normalized mixture coefficient.

After initializing the coefficients in Equation (11), we use Expectation-Maximization for solving. By alternately executing these two steps until the function converges, we can learn K Gaussian components. Their covariance matrices represent feature information. The image processing procedure is shown in Fig. 3.

Fig. 3 Processing diagram of Image Gaussian distribution

Next, we can use the learned prior information to substitute into the group sparse model in Equation (10) and conditionally relax the sampling operator Φ. We consider the model's block coherence and within-group correlation v(Φ):

(7)

, (8)

where describes the global properties of Φ, including the local similarity and smoothness of the remote sensing image. is the qth column vector in . According to Eldar and Kutyniok (2012), Φ only needs to satisfy Equation (14) to correctly perceive the original signal X:

. (9)

When , the perception process can be converted into solving the following mixed norm minimization problem:

(10)

;

when q=1, it is the group sparse-induced norm. Through iterative shrinkage-thresholding methods, we can minimize the mixed norm, promoting simultaneous zeros within each group to achieve group sparsity.

Thus, we have demonstrated that the proposed group sparse model can effectively perform compressive sampling of the original signal and achieve high-probability recovery within the compressive sensing. Fig. 4 illustrates the difference between our group sparse sampling sequence and the regular sampling sequence.

Fig. 4 Diagram of Group Sparse Sampling Sequence and Regular Sampling Sequence

3.3 Channel self-decay normalization based on low-order statistics

Unlike CNNs, which can process arbitrary inputs using a sliding window approach, SNNs require neurons to generate spike sequences based on input magnitude. In this context, weights and threshold voltages are responsible for ensuring the adequacy and balance of neuron activation, respectively. However, during processing, neurons may become under-activated or over-activated, leading to information loss and poor performance. Therefore, in SNNs, we apply appropriate normalization techniques to the input signals. Common normalization methods include batch normalization (BN) and layer normalization (LN). BN, originally designed for deep ANNs, can also be adapted for use in SNNs. By normalizing the input features across the entire batch, BN facilitates faster convergence and may improve generalization. LN, on the other hand, normalizes the inputs across all units within a single sample, making it particularly useful for handling variable-length sequence data. However, when addressing normalization errors, spiking-YOLO identified that layer-norm significantly reduced neuron firing rates, causing underactivation. Therefore, spiking-YOLO employed fine-grained channel normalization to ensure that even minimal activation values are properly normalized. The channel normalization formula in spiking-YOLO is as follows:

(11)

However, further analysis revealed that while channel normalization addressed the underactivation issue, it did not resolve—and even exacerbated—the overactivation problem. This was caused by spikes of inactivated neurons, known as the SIN error. In fact, the SIN error already exists with the more commonly used layer-norm in image classification. Since current conversion used real values for input, assuming no neuron activation values exceed 1, the firing rate output by the encoding layer, receiving a constant current over sufficient time steps, can exactly match the ANN's activation value. This can be expressed as:

, (12)

where is the synaptic current and is the activation function value.

After the first layer, neurons receive discrete spikes. According to the literature (Li, et al., 2022), disregarding the error from the floor function, the number of emitted spikes is the time-varying maximum of the synaptic current sum for each layer's neurons. Thus, the total number of spikes

over time steps T equals .

When the activation function value of the i^{th} neuron in the K^{th} layer is less than 0, but the synaptic current exceeds the threshold, the neuron will still fire spikes even though it should not be activated in the ANN. This results in the SIN error. The spike error in the K^{th} layer can be expressed as:

. (13)

Since the firing rate in SNNs provides high variance while the activation function values in ANNs provide a high mean, the mathematical distributions of their hidden layers differ. Additionally, similar to error accumulation in ANNs, the SIN problem becomes increasingly pronounced with more layers and channels. This issue is particularly significant for neural networks dealing with remote sensing images and cannot be ignored. For example, R3Det has 4096 channels, and the accumulation of errors across these numerous channels significantly increases the conversion error in the resulting SNN. To address this issue, spike calibration (Li et al., 2022) was proposed, using a spike monitor to calculate the interspike interval (ISI) and using neurons with negative weights to counteract the activation of inactive neurons. However, this method increased the number of neurons unnecessarily, leading to additional computational overhead.

To address the SIN error in a cost-effective manner, we proposed an exponential momentum decay scheme based on low-order statistics, building on the existing channel normalization in spiking-YOLO, as shown in Fig. 5. This method limits excessive neuron firing rates. We denoted the variance and mean of neuron activation in the k^{th} layer and c^{th} channel as and , respectively. Introducing a momentum decay factor , the mean and variance are progressively updated as the layer's depth increases:

(14)

(15)

where is the object momentum coefficient with adaptive decay, and and are the mean and variance of the current channel normalization, respectively.

In light of this, as the network depth increases, the normalized weights gradually decrease. This reduction is not abrupt but occurs in a smooth manner, making the process more biologically plausible. Our method effectively suppresses abnormally high firing rates of neurons, leading to more faithful information processing and reduced unnecessary energy consumption. Importantly, we achieve this by only adjusting the weight decay parameters. This approach eliminates overactivated neurons without significantly increasing the computational burden. Consequently, our technique is both efficient and practical, making it well-suited for large-scale network deployments in real-world applications.

Fig. 5 Diagram of exponential momentum attenuation scheme

4 Experiments

We evaluated the performance of the proposed method on object detection, using R3Det as the initial algorithm before conversion. The subsequent SNN conversion was performed by the SNN Toolbox. In addition to standard model parsing, normalization, and simulation procedures, we implemented the rapid perception model and channel self-decay normalization on the Toolbox platform. Max pooling and batch normalization were implemented according to Rueckauer et al. (2017) We conducted our experimental analysis using two publicly available datasets annotated with oriented bounding boxes: DOTA (Xia et al., 2018) and HRSC2016 (Liu et al., 2016). The experimental setup is detailed in Table 1. In inference, we use rate coding to encode the input images to the input spikes by the IF neurons.

Table 1 Experimental environment.

Category	Version
Server	Inspur NF5466M5
CPU	Intel(R) Xeon(R)
GPU	NVIDIA Tesla V100
Memory	253GiB
OS	Debian 11 Bullseye
SNN Framework	SNNToolbox (Rueckauer and Liu, 2021)
Language	Python3.7

4.1 Datasets and settings

DOTA is a large open remote sensing image benchmark dataset comprising thousands of images from different platforms and sensors such as Google Earth, the GF-2 and JL-1 satellites, etc. DOTA-v1.0 contains 2806 images ranging in size from 800*800 pixels to 4000*4000 pixels, and the dataset is divided into a training set, validation set, and test set in the ratio of 3:1:2. The images cover 15 object types with a total of 188,282 objects, including helicopter (HC), swimming pool (SP), harbor (HA), traffic circle (RA), soccer field (SBF), storage tank (ST), basketball court (BC), tennis court (TC), ship (SH), large vehicle (LV), small vehicle (SV), track and field (GTF), bridge (BR), baseball fields (BD), and aircraft (PL). We trained the model for 36 rounds (183,600 iterations).

The high-resolution ship collections 2016 (HRSC2016) dataset is mainly used for various ship detection tasks. It labels three major categories of ship: aircraft carrier, warcraft, and merchant ship. Within the three categories, there are 27 sub-categories of objects, with a total of 2976 objects. The training set includes 436 images with 1207 samples, the validation set includes 181 images with 541 samples, and the test set includes 444 images with 1228 samples.

4.2 Efficiency of S3Det

To validate and analyze the efficiency of the proposed method, we examined the impact of the SSRS module and CSWN on both performance and energy consumption.

4.2.1 Detection efficiency

First, we evaluated detection efficiency, as image processing speed is a crucial metric for real-time applications on embedded devices such as drones. We compared S3Det's speed with two categories of algorithms: common remote sensing image detection algorithms and lightweight detection algorithms suitable for rotated objects. Additionally, to enhance detection speed on edge devices, we replaced S3Det's backbone with MobileNetV2 (Sinha and El-Sharkawy, 2019) and ShuffleNetV2 (Ma et al., 2018b) and measured the relevant performance indicators.

"#Params" represents the total number of model parameters. "Ratio" indicates the proportion of the actual runtime consumed by the model in relation to the overall inference time taken to process 1000 sub-images. This ratio is computed using the standardized method provided by torchstat. Meanwhile, the frames per second (FPS) are measured using MMDetection (Chen et al., 2019), and the testing environment comprises four Tesla V100 GPUs with a batch size set to 16. The detection results of the algorithm are shown in Table 2.

Table 2 Speed comparison of DOTA and HRSC2016. The speed of R2CNN is <1fps, so it is indicated by "-"

Model	Backbone	Image Size	DOTA			HRSC2016		
			#Params	Ratio	Speed	#Params	Ratio	Speed
R^3Det(Yang, et al., 2021)	ResNet50	800*800	485MiB	88.52%	14 fps	496MiB	89.42%	8fps
SCRDet(Yang et	ResNet50	800*800	452MiB	71.20%	10 fps	484MiB	76.03%	4.5fps

al., 2019)									
R^2CNN(Jiang et al., 2017)	ResNet50	600*600	353MiB	93.60%	-	314MiB	94.41%	2fps	
RRPN(Ma et al., 2018a)	ResNet50	600*600	348MiB	94.30%	5fps	306MiB	92.43%	3.5fps	
RetinaNet-R(Tsung-Yi et al., 2017)	ResNet50	800*800	378MiB	82.85%	12fps	366MiB	83.90%	10fps	
MobileDet-R(Xiong et al., 2021)	MobileNetV3	300*300	96MiB	31.78%	41.5fps	104MiB	35.98%	33fps	
GiraffeDet-R(Jiang et al., 2022)	S2D Chain	300*300	137MiB	35.32%	35fps	153MiB	53.64%	24fps	
OR-CNN(Xie et al., 2024c)	ResNet50	800*800	368MiB	37.31%	15.3fps	269MiB	78.43%	14.9fps	
DFDet(Xie et al., 2024b)	ResNet50	800*800	392MiB	36.82%	23.4fps	280MiB	84.52%	21fps	
YOLOv10-L(Wang et al., 2024)	CSPNet	300*300	152MiB	40.73%	32fps	96MiB	31.22%	32fps	
S3Det（Step=64）	ResNet50	800*800	**462MiB**	**64.47%**	**20fps**	**482MiB**	**66.59%**	**14fps**	
	ResNet101	800*800	758MiB	72.57%	12fps	684MiB	77.22%	8fps	
	ResNet152	800*800	961MiB	85.29%	8fps	887MiB	89.17%	6fps	
	MobileNetV2	300*300	168MiB	32.14%	35fps	121MiB	33.98%	31fps	
	ShuffleNetV2	300*300	**147MiB**	**30.47%**	**41fps**	**107MiB**	**32.10%**	**34fps**	

As shown in Table 2, using the ResNet as the backbone, S3Det with a stride of 64 significantly improves detection speed (FPS) compared to the ANN model (R3Det). Specifically, using ResNet-50 as the backbone network, the detection speed of S3Det is 20 FPS, representing a 42.86% increase over R3Det. Additionally, by using a more lightweight backbone. such as ShuffleNet, the detection speed can be rapidly increased to 41 FPS, further demonstrating S3Det's efficiency. We also noted that the inference time ratio of the S3Det model represented a significant improvement over that of the baseline model (DOTA: 64.47% vs. 88.52%, HRSC2016: 66.59% vs. 89.42%). However, the model's parameter count did not show a substantial decrease. This may be attributed to the SNN model parameters being largely inherited from the ANN, with our SSRS primarily reducing input redundancy, resulting in no substantial reduction in the parameter count.

4.2.2 Energy efficiency

SNNs exhibit exceptionally low power consumption due to their event-driven neural activity and rich spatio-temporal dynamics. In our approach, the sparsity gain introduced by the SSRS further enhances the optimization of the model's operational power consumption. To thoroughly understand the role of sparsity in SNN, we evaluated the power consumption metrics of the R3Det model before conversion and the S3Det model after conversion. Before conversion, we divided R3Det into four sub-networks: ResNet, FPN, RPN, and RefineNet. Fig. 6 presents the number of multiply-add for each sub-network. The total MACs for all networks amount to 216.7 GMACs.

Fig. 6 Number of MAC operations of the sub-network

For the post-conversion, we define one time step as 1 ms (1 kHz synchronization signal in Merolla et al. [2014]). According to Horowitz et al. (2014), the energy cost per operation is 4.6 pJ (FLOAT32 MAC) and 0.9 pJ (FLOAT32 AC). Considering that S3Det accepts analog image inputs, we defined the first layer as MAC operations. Rathi and Roy (2023) argue that the ratio of ANN to SNN energy consumption can be expressed as:

(16)

Since S3Det was converted from the ANN network R3Det, we calculated the energy consumption of S3Dets using the following formula:

, (17)

where E_{AC} denotes the energy consumption of a floating-point plus.

We recorded the number of SNN operations and the average spike rate during S3Det's inference over a specific time step. To facilitate comparison, we included R3Det's power consumption metrics and calculated the detailed energy consumption, as summarized in Table 3.

Table 3 Energy consumption comparison on DOTA. In SNNs, model complexity is measured using spike operations and firing rates. Therefore, the FLOPs (Floating Point Operations per Second) metric is not applicable and is left as "--" in the table. In contrast, for ANNs, FLOPs remain the standard metric for measuring computational complexity

	Data Type	Input	FLOPs	OP_{S3Det}	Spiking Rate	Energy	Power(W)
R3Det	Float 32bit	800×800	4.334E+11	--	--	1.994	178
S3Det(Channel Norm)T=64	Float 32bit	800×800	--	4.275E+11	39.76%	1.53E-01	2.39
S3Det(Channel self-decay norm)T=64	Float 32bit	800×800	--	4.275E+11	24.32%	9.36E-02	1.46

Our calculations in Table 3 indicate that using CSWN, the S3Det model consumed approximately 21 times less energy and about 122 times less power than R3Det when the input was a 32-bit floating-point. This demonstrates the significant low-power advantage of S3Det. Additionally, our proposed channel self-decay normalization method reduced the spike rate by 15.44%, effectively addressing the issue of abnormal neuron activation and limiting excessive firing rates. This reduction in spike rate further contributed to the overall decrease in the model's power consumption.

4.3 High-precision detection experiment

Recent studies (Li, et al., 2022; Hu, et al., 2023) have shown that converted SNNs can match or even exceed the performance of ANNs in natural image object detection. However, these algorithms have not performed as well on remote sensing data. In our accuracy experiment, we utilized average precision (AP) and mean average precision (mAP), which are standard metrics in object detection. In this section, we presented the results of S3Det on DOTA and HRSC2016. As a benchmark, we conducted comparative experiments of R3Det and S3Det using ResNet-50, ResNet-101, and ResNet-152. In addition, we evaluated the detection accuracy of several commonly used algorithms.

On DOTA, our proposed spiking conversion method is based on the R3Det benchmark. Hence, we selected comparison methods that have frequently been used with R3Det in recent studies, including high-precision algorithms such as one-stage detectors DAL (Ming et al., 2021) and S2ANet (Jiaming et al., 2020), and two-stage detectors OR-CNN (Xie, et al., 2024c) and DFDet (Xie, et al., 2024b). On HRSC2016, RC1 (Liu, et al., 2016) and RRD (Yang and Yan, 2022) used VGG16 as the backbone, while the remaining algorithms utilized ResNet101. Different methods use various input image sizes. To highlight the efficiency advantages, we did not use the most accurate backbone networks for the comparison, as this would significantly increase the detection time with only marginal gains in accuracy. Instead, we opted for typical configurations to provide a fair and practical comparison. The experimental results are shown in Tables 4 and 5.

Table 4 Evaluation of the OBB task on the DOTA testing set. The bolded numbers indicate the highest accuracy rate under the current category

Method	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
One-stage																
RetinaNet-R	88.92	67.67	33.55	56.83	66.11	73.28	75.24	**90.87**	73.95	75.07	43.77	56.72	51.05	55.86	21.46	62.02
DAL(Ming, et al., 2021)	88.68	76.55	45.08	66.80	67.00	76.76	79.74	90.84	79.54	78.45	57.71	62.27	69.05	73.14	60.11	71.44
S^2ANet(Jiaming, et al., 2020)	89.11	**82.84**	48.37	71.11	78.11	78.39	87.25	90.83	84.90	85.64	60.36	62.60	65.26	69.13	57.94	74.12
R^3Det-50	89.30	80.29	46.21	65.07	70.51	73.38	77.42	90.83	80.59	82.26	59.29	58.25	57.75	65.90	55.31	70.16
R^3Det-101	89.54	81.99	48.46	62.52	70.48	74.29	77.54	90.80	81.39	83.54	61.97	59.82	65.44	67.46	60.05	71.69
R^3Det-152	89.42	81.03	50.41	65.93	70.90	**78.63**	78.03	90.67	85.24	84.10	61.64	63.52	68.15	69.80	**67.09**	73.63
Two-stage																
SCRDet	**89.98**	80.65	**52.09**	68.36	64.52	60.32	72.41	90.85	**87.94**	**86.86**	**65.02**	66.68	66.25	68.24	65.21	72.36
R^2CNN	80.94	65.67	35.34	67.44	59.92	50.91	55.81	90.67	66.92	72.39	55.06	52.23	55.14	53.35	48.22	60.67
RRPN	88.52	71.2	31.66	59.3	51.85	56.19	57.25	**90.81**	72.84	67.38	59.69	52.84	53.08	51.94	53.58	61.21
ICN(Azimi et al., 2018)	81.36	74.30	47.70	70.32	64.89	67.82	69.98	90.76	79.06	78.20	53.64	62.90	67.02	64.17	50.23	68.16
CAD-Net(Gongjie et al., 2019)	87.80	82.40	49.40	**73.50**	71.10	63.50	76.60	90.90	79.20	73.30	48.40	60.90	62.00	67.00	62.20	69.90
OR-CNN	89.46	82.12	54.78	70.86	**78.93**	83.00	**88.20**	90.90	87.50	84.68	63.97	**67.69**	**74.94**	68.84	52.28	**75.87**
DFDet	88.92	79.25	48.40	70.00	80.22	78.85	87.21	90.90	83.13	83.98	60.07	66.49	68.27	**76.78**	58.22	74.71
Ours(Step=512)																
S3Det-50	87.83	77.45	34.05	64.50	62.34	73.10	66.11	90.56	75.33	78.28	55.37	56.12	62.88	64.11	52.84	66.72
S3Det-101	89.16	77.79	43.62	58.11	66.56	70.99	72.22	86.89	77.34	79.36	57.41	55.11	60.90	63.78	56.89	67.74
S3Det-152	88.92	80.10	48.01	62.65	70.31	71.48	72.23	85.95	80.77	81.89	60.44	56.00	65.36	67.80	63.11	**70.33**

Table 5 Accuracy and speed comparison on HRSC2016. The speed of RC1 and RRD is <1 fps, so it is indicated by "-"

Method	Backbone	Image Size	mAP(%)	Speed(fps)
R^2CNN	ResNet101	600×600	73.07	2.4
RRPN	ResNet101	600×600	79.08	3.7
RC1	VGG16	800×800	75.71	-
RRD	VGG16	384×384	84.32	-
RoI-Transformer (Ding et al., 2018)	ResNet101	512×800	86.24	6.1
R^3Det	ResNet101	800×800	**89.26**	10.6
S3Det (Step=512)	ResNet101	800×800	**85.24**	**12.4**

The accuracy experiments revealed that the S3Det achieved a strong detection performance on both DOTA and HRSC datasets. With a stride of 512 and ResNet-152 as the backbone, S3Det attained 70.33% mAP on DOTA, which is only a 3.3% reduction compared to the original algorithm (73.63%). While the theoretical conversion from ANN to SNN is designed to be lossless, a significant mismatch between the firing rates and the activation values persists. This discrepancy often leads to the converted SNNs achieving lower detection accuracy compared to the ANNs. However, other than in scenarios demanding extremely high precision, the detection performance of S3Det is suitable for most tasks. Thus, the accuracy loss is deemed acceptable. The detection results are visualized in Fig. 7. We also plotted the AP change curves for each category at different time steps, and, as shown in Fig.8, the APs of all categories are positively correlated with the growth of T and eventually converge to a threshold value.

Fig. 7 Visualization of S3Det (T=512) on DOTA. The short names for categories are defined as: SV, Small vehicle; LV, Large vehicle; SH, Ship; HC, Helicopter; PL, Plane; TC, Tennis court; BR, Bridge; SP, Swimming pool; RA, Roundabout; HB, Harbor; BD, Baseball diamond; GTF, Ground field track; SBF, Soccer-ball field; ST, Storage tank; and BC, Basketball court

Fig. 8 The mAP for each category at different time steps

For the HRSC2016 dataset, we conducted comparative validation of our method's detection performance by varying time steps. Fig. 9 demonstrates the detection advantages of our method as the time steps increase. When T=256, S3Det successfully detected three types of ships, whereas the Spiking-R3Det, converted directly from R3Det, only succeeded at 1280 steps. Additionally, at this time step, the bounding box for the Merchant ship (colored pink) was inaccurately drawn. We hypothesize that this is due to the group sparse model, which sparsifies the object pixels, thereby reasonably reducing the processing time occupied by redundant pixels. Additionally, the number of invalid boxes produced by the channel norm is significantly higher than that by the channel self-decaying norm. We believe this is because the channel self-decaying norm effectively suppresses the SIN error and mitigates the abnormal spiking, thereby reducing the issue of erroneous pixel detection.

Fig. 9 Spiking-R3Det and S3Det visualization results on HRSC2016. T denotes time step, pink represents a Merchant ship, green represents an Aircraft carrier, blue represents Warcraft.

5 Conclusions

In this work, we developed a comprehensive and efficient conversion framework for detecting remote sensing images while maintaining low power consumption. Our method achieved high precision with fewer time steps compared to benchmark methods, while maintaining significantly lower power consumption than ANN.

In fact, our conversion method can be applied to almost all ANNs. Experimental results from object detection tasks demonstrate that our conversion method can significantly reduce energy consumption while maintaining or even improving performance. To evaluate the versatility across different scenarios, we are currently testing the converted SNNs in various settings, including real-time mobile applications, embedded systems, and edge computing environments. That being said, the conversion process may not always preserve the exact behavior of the original ANN, especially when the ANN relies heavily on fine-tuned weights and complex nonlinearities. Additionally, the conversion process may introduce a small overhead in terms of computational resources during the initial setup, which could be a consideration for extremely resource-constrained applications. To further enhance the generalizability of our method, future work will focus on direct training and inference of SNNs.

Contributons

Li Chen was responsible for the conception and planning of the experimental ideas, and led the design of the experiments. He also participated in data analysis and interpretation of results. Fan Zhang undertook the task of constructing the theoretical model and conducted a comprehensive review of the relevant literature. Guangwei Xie focused on the data collection process and the execution of the experiments. YanZhao Gao was in charge of organizing the theoretical knowledge into mathematical formulations. Xiaofeng Qi took the lead in drafting the initial manuscript and subsequent revisions, ensuring that the paper's structure was logical and the language was fluent. Mingqian Sun managed the collection and organization of preliminary research materials.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Azimi SM, Vig E, Bahmanyar R, et al., 2018. Towards multi-class object detection in unconstrained remote sensing imagery. Proc 14th Asian Conf on Computer Vision, p.150-165. https://doi.org/10.1007/978-3-030-20893-6_10

Chen GH, Pei GS, Tang Y, et al., 2022. A novel multi-sample data augmentation method for oriented object detection in remote sensing images. Proc IEEE 24th Int Workshop on Multimedia Signal Processing, p.1-7. https://doi.org/10.1109/MMSP55362.2022.9949615

Chen K, Wang JQ, Pang JM, et al., 2019. MMDetection: open MMlab detection toolbox and benchmark. https://arxiv.org/abs/1906.07155.

Chen L, Zhang F, Guo W, et al., 2023. SFTN: fast object detection for aerial images. *IET Image Process*, 17(13):3897-3907. https://doi.org/10.1049/ipr2.12906

Cheng G, Zhou PC, Han JW, et al., 2016a. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans Geosci Remote Sensing*, 54(12):7405-7415. https://doi.org/10.1109/TGRS.2016.2601622

Cheng G, Zhou PC, Han JW, 2016b. RIFD-CNN: rotation-invariant and fisher discriminative convolutional neural networks for object detection. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.2884-2893. https://doi.org/10.1109/CVPR.2016.315

Ding J, Xue N, Long Y, et al., 2019. Learning RoI transformer for oriented object detection in aerial images. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.2844-2853. https://doi.org/10.1109/CVPR.2019.00296

Eldar YC, Kutyniok G, 2012. Compressed Sensing: Theory and Applications. Cambridge University Press, Cambridge, UK. https://doi.org/10.1017/CBO9780511794308

Everingham M, Van Gool L, Williams CK, et al., 2010. The PASCAL visual object classes (VOC) challenge. *Int J Comput Vision*, 88(2):303-338. https://doi.org/10.1007/s11263-009-0275-4

Gong MG, Li JZ, Zhang YR, et al., 2022. Two-path aggregation attention network with quad-patch data augmentation for few-shot scene classification. *IEEE Trans Geosci Remote Sensing*, 60:4511616. https://doi.org/10.1109/TGRS.2022.3197445

Han JM, Ding J, Li J, et al., 2022. Align deep features for oriented object detection. *IEEE Trans Geosci Remote Sensing*, 60:5602511. https://doi.org/10.1109/TGRS.2021.3062048

He X, Ma SP, He LY, et al., 2022. High-resolution polar network for object detection in remote sensing images. *IEEE Geosci Remote Sensing Lett*, 19:6000605. https://doi.org/10.1109/LGRS.2020.3039240

Horowitz M, 2014. 1.1 Computing's energy problem (and what we can do about it). Proc IEEE Int Solid-State Circuits Conf Digest of Technical Papers, p.10-14. https://doi.org/10.1109/ISSCC.2014.6757323

Hu YF, Zheng Q, Jiang XD, et al., 2023. Fast-SNN: fast spiking neural network by converting quantized ANN. *IEEE Trans Pattern Anal Mach Intell*, 45(12):14546-14562. https://doi.org/10.1109/TPAMI.2023.3275769

Huang ZC, Li W, Xia XG, et al., 2022. A general gaussian heatmap label assignment for arbitrary-oriented object detection. *IEEE Trans Image Process*, 31:1895-1910. https://doi.org/10.1109/TIP.2022.3148874

Jiang YQ, Tan ZY, Wang JY, et al., 2022. GiraffeDet: a heavy-neck paradigm for object detection. https://arxiv.org/abs/2202.04256

Jiang YY, Zhu XY, Wang XB, et al., 2018. R^2CNN: rotational region CNN for arbitrarily-oriented scene text detection. Proc 24th Int Conf on Pattern Recognition, p.3610-3615. https://doi.org/10.1109/ICPR.2018.8545598

Kim S, Park S, Na B, et al., 2020. Spiking-YOLO: spiking neural network for energy-efficient object detection. Proc 34th AAAI Conf on Artificial Intelligence, p.11270-11277. https://doi.org/10.1609/aaai.v34i07.6787

Komárek A, Lesaffre E, 2008. Generalized linear mixed model with a penalized gaussian mixture as a random effects distribution. *Comput Stat Data Anal*, 52(7):3441-3458. https://doi.org/10.1016/j.csda.2007.10.024

Lecun Y, Bottou L, Bengio Y, et al., 1998. Gradient-based learning applied to document recognition. *Proc IEEE*, 86(11):2278-2324. https://doi.org/10.1109/5.726791

Li Y, He X, Dong YT, et al., 2022. Spike calibration: fast and accurate conversion of spiking neural network for object detection and segmentation. https://arxiv.org/abs/2207.02702

Lin TY, Maire M, Belongie S, et al., 2014. Microsoft coco: common objects in context. Proc 13th European Conf on Computer Vision, p.740-755. https://doi.org/10.1007/978-3-319-10602-1_48

Lin TY, Goyal P, Girshick R, et al., 2017. Focal loss for dense object detection. Proc IEEE Int Conf on Computer Vision, p.2999-3007. https://doi.org/10.1109/ICCV.2017.324

Liu WX, Luo B, Liu J, et al., 2022. Synthetic data augmentation using multiscale attention cyclegan for aircraft detection in remote sensing images. *IEEE Geosci Remote Sensing Lett*, 19:4009205. https://doi.org/10.1109/LGRS.2021.3052017

Liu ZK, Wang HZ, Weng LB, et al., 2016. Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds. *IEEE Geosci Remote Sensing Lett*, 13(8):1074-1078. https://doi.org/10.1109/LGRS.2016.2565705

Ma JQ, Shao WY, Ye H, et al., 2018. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans Multimedia*, 20(11):3111-3122. https://doi.org/10.1109/TMM.2018.2818020

Maass W, 1997. Networks of spiking neurons: the third generation of neural network models. *Neural Networks*, 10(9):1659-1671. https://doi.org/10.1016/S0893-6080(97)00011-7

Merolla PA, Arthur JV, Alvarez-Icaza R, et al., 2014. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668-673. https://doi.org/10.1126/science.1254642

Ming Q, Zhou ZQ, Miao LJ, et al., 2021. Dynamic anchor learning for arbitrary-oriented object detection. Proc 35th AAAI Conf on Artificial Intelligence, Electr Network. p.2355-2363. https://doi.org/10.1609/aaai.v35i3.16336

Rathi N, Roy K, 2023. DIET-SNN: a low-latency spiking neural network with direct input encoding and leakage and threshold optimization. *IEEE Trans Neural Netw Learning Syst*, 34(6):3174-3182. https://doi.org/10.1109/TNNLS.2021.3111897

Rueckauer B, Liu SC, 2021. Temporal pattern coding in deep spiking neural networks. Proc Int Joint Conf on Neural Networks, p.1-8. https://doi.org/10.1109/IJCNN52387.2021.9533837

Rueckauer B, Lungu IA, Hu YH, et al., 2017. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Front Neurosci*, 11:682. https://doi.org/10.3389/fnins.2017.00682

Sinha D, El-Sharkawy M, 2019. Thin MobileNet: an enhanced mobilenet architecture. Proc IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conf, p.280-285. https://doi.org/10.1109/UEMCON47517.2019.8993089

Vaswani A, Shazeer N, Parmar N, et al., 2017. Attention is all you need. Proc 31st Int Conf on Neural Information Processing Systems, p.6000-6010.

Wang A, Chen H, Liu LH, et al., 2024. YOLOv10: real-time end-to-end object detection. https://arxiv.org/abs/2405.14458

Xia GS, Bai X, Ding J, et al., 2018. DOTA: a large-scale dataset for object detection in aerial images. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.3974-3983. https://doi.org/10.1109/CVPR.2018.00418

Xie XX, Lang CB, Miao SC, et al., 2023. Mutual-assistance learning for object detection. *IEEE Trans Pattern Anal Mach Intell*, 45(12):15171-15184. https://doi.org/10.1109/TPAMI.2023.3319634

Xie XX, Cheng G, Li QY, et al., 2024a. Fewer is more: efficient object detection in large aerial images. *Sci China Inf Sci*, 67(1):112106. https://doi.org/10.1007/s11432-022-3718-5

Xie XX, Cheng G, Rao CF, et al., 2024b. Oriented object detection via contextual dependence mining and penalty-incentive allocation. *IEEE Trans Geosci Remote Sensing*, 62:5618010. https://doi.org/10.1109/TGRS.2024.3385985

Xie XX, Cheng G, Wang JB, et al., 2024c. Oriented R-CNN and beyond. *Int J Comput Vision*, 132(7):2420-2442. https://doi.org/10.1007/s11263-024-01989-w

Xiong YY, Liu HX, Gupta S, et al., 2021. MobileDets: searching for object detection architectures for mobile accelerators. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.3824-3833. https://doi.org/10.1109/CVPR46437.2021.00382

Yang X, Yan JC, 2022. Correction to: on the arbitrary-oriented object detection: classification based approaches revisited. *Int J Comput Vision*, 130(7):1873-1874. https://doi.org/10.1007/s11263-022-01618-4

Yang X, Yang JR, Yan JC, et al., 2019. SCRDet: towards more robust detection for small, cluttered and rotated objects. Proc IEEE/CVF Int Conf on Computer Vision, p.8231-8240. https://doi.org/10.1109/ICCV.2019.00832

Yang X, Yan JC, Feng ZM, et al., 2021a. R3Det: refined single-stage detector with feature refinement for rotating object. Proc 35th AAAI Conf on Artificial Intelligence, p.3163-3171. https://doi.org/10.1609/aaai.v35i4.16426

Yang X, Yan JC, Ming Q, et al., 2021b. Rethinking rotated object detection with Gaussian wasserstein distance loss. Proc 38th Int Conf on Machine Learning, p.11830-11841.

Yao M, Zhao GS, Zhang HY, et al., 2023. Attention spiking neural networks. *IEEE Trans Pattern Anal Mach Intell*, 45(8):9393-9410. https://doi.org/10.1109/TPAMI.2023.3241201

Zhang C, Lam KM, Wang Q, 2023. CoF-Net: a progressive coarse-to-fine framework for object detection in remote-sensing imagery. *IEEE Trans Geosci Remote Sensing*, 61:5600617. https://doi.org/10.1109/TGRS.2022.3233881

Zhang GJ, Lu SJ, Zhang W, 2019. CAD-Net: a context-aware detection network for objects in remote sensing imagery. *IEEE Trans Geosci Remote Sensing*, 57(12):10015-10024. https://doi.org/10.1109/TGRS.2019.2930982