SpikingYOLOX: Improved YOLOX Object Detection with Fast Fourier Convolution and Spiking Neural Networks

Wei Miao[*1, 2], **Jiangrong Shen**[*3, 4], **Qi Xu**[1†], **Timo Hamalainen**[2], **Yi Xu**[5], **Fengyu Cong**[6]

[1]School of Computer Science and Technology, Dalian University of Technology,
[2]Faculty of Information Technology, University of Jyväskylä,
[3]School of Computer Science and Technology, Xi'an Jiaotong University,
[4]State Key Lab of Brain-Machine Intelligence, Zhejiang University,
[5]School of Control Science and Engineering, Dalian University of Technology,
[6]School of Biomedical Engineering, Dalian University of Technology
weimiao@jyu.fi, jrshen@zju.edu.cn, xuqi@dlut.edu.cn, timo.t.hamalainen@jyu.fi, yxu@dlut.edu.cn, cong@dlut.edu.cn

Abstract

In recent years, with the advancements in brain science, spiking neural networks (SNNs) have garnered significant attention. SNNs can generate spikes that mimic the function of neurons transmission in humans brain, thereby significantly reducing computational costs by the event-driven nature during training. While deep SNNs have shown impressive performance on classification tasks, they still face challenges in more complex tasks such as object detection. In this paper, we propose SpikingYOLOX, extending the structure of the original YOLOX by introducing signed spiking neurons and fast Fourier convolution (FFC). The designed ternary signed spiking neurons could generate three kinds of spikes to obtain more robust features in the deep layer of the backbone. Meanwhile, we integrate FFC with SNN modules to enhance object detection performance, because its global receptive field is beneficial to the object detection task. Extensive experiments demonstrate that the proposed SpikingYOLOX achieves state-of-the-art performance among other SNN-based object detection methods.

Introduction

Significant advancements in deep learning have greatly enhanced the performance of object detection algorithms in terms of both accuracy and real-time processing. Object detection focuses on identifying and locating multiple overlapping objects within an image, providing precise bounding boxes for each object. Techniques based on deep networks, such as Convolutional Neural Networks (CNNs), Region-based CNNs (R-CNN), and single-shot detectors like YOLO (You Only Look Once), have been widely adopted. Notably, the YOLO series (Redmon et al. 2016; Redmon and Farhadi 2017, 2018; Bochkovskiy, Wang, and Liao 2020; Jocher 2020; Li et al. 2022a; Ge et al. 2021; Wang, Bochkovskiy, and Liao 2023) has consistently pursued optimal speed and accuracy for real-time applications. These methods have demonstrated remarkable potential and applicability across various domains, including autonomous driving, surveillance, and medical imaging analysis. However, as data volumes increase and model sizes grow, there is an urgent need to develop low-energy consumption object detection models. Spiking neural networks (SNNs), regarded as potential competitors to artificial neural networks (ANNs) for their high biological plausibility and low power consumption (Rueckauer et al. 2017; Tavanaei et al. 2019; Xu et al. 2018, 2023), offer promising potential for implementing energy-efficient object detection models.

Recently, SNN-based techniques have been applied to object recognition and detection, achieving performance close to that of ANNs on simpler classification datasets (Tavanaei et al. 2019). To bridge the performance gap between ANNs and SNNs, deeper network structures within SNNs, such as ResNet, have been explored (Hwang et al. 2021; Hu, Tang, and Pan 2021; Zheng et al. 2021; Samadzadeh et al. 2023). Spiking-YOLO (Kim et al. 2020b) represents a pioneering effort in this area, implementing channel-wise data-based normalization to mitigate information loss; however, this approach led to significant performance degradation. To further narrow the gap between ANN- and SNN-based YOLO models in object detection, EMS-YOLO (Su et al. 2023) was the first to apply a trained Spiking ResNet to object detection. These studies have demonstrated the high-performance potential of SNNs in object detection, as well as their low energy consumption. Nevertheless, when faced with more complex datasets, such as those found in the COCO dataset, SNNs still significantly lag behind ANNs in object detection scenarios. Therefore, we focus on how to improve the object detection by using SNN.

One of the bottlenecks impacting the network's ability to extract features when using SNN on object detection is the size and structure of the receptive field. In the context of convolutional kernel operations, which are commonly used to extract local features in object detection tasks, the receptive field refers to the region of the input image that a particular neuron is "looking at." As these operations are applied in deep convolutional neural networks, successive convolu-

[*]These authors contributed equally.
[†]Corresponding author : Qi Xu, xuqi@dlut.edu.cn

tional and pooling layers progressively enlarge the receptive field, allowing the network to capture more comprehensive features from the input image. A large effective receptive field is essential for understanding the global structure of an image and for solving object detection tasks effectively. However, we observe that many current popular SNNs lack a sufficiently large receptive field, even with deeper layer structures. (Xu et al. 2022; Shen et al. 2023) To address this limitation and achieve a global receptive field, spectral operations present a viable solution. Therefore, we introduce Fast Fourier Convolutions (FFCs), which provide a receptive field encompassing the entire image. We demonstrate that this property of FFCs significantly enhances object detection performance.

In this paper, we propose SpikingYOLOX, marking the first application of SNNs based on the YOLOX architecture. We integrate spiking neurons in the deeper layers for feature extraction to implement low computational cost. To enhance performance, we design a tenary signed spiking neuron, inspired by fastSNN (Hu et al. 2023), to replicate the functionality of activation functions in ANNs. Additionally, we employ FFCs in the bottleneck part of the backbone, leveraging their global receptive field for superior feature extraction. We evaluate our model on the COCO2017 dataset, and our experiments demonstrate that both the signed spiking neurons and the application of FFCs significantly improve object detection performance. To the best of our knowledge, this is the first application of an SNN module to the YOLOX architecture, trained in an end-to-end manner. The contributions of this paper are as follows:

- We propose SpikingYOLOX, an innovative SNN object detection model. This represents the first instance of integrating SNNs into the YOLOX architecture, demonstrating that SNNs can achieve high-performance object detection tasks with low energy consumption.
- We introduce two novel SNN modules: CSP-FFC-SNN and SPP-SNN, to harness the strengths of both ANNs and SNNs. The CSP-FFC-SNN module incorporates FFC to provide a global receptive field, enhancing feature extraction and improving object detection performance. The SPP-SNN module integrates Spatial Pyramid Pooling (SPP) with spiking neurons to further refine detection capabilities.
- We propose a ternary signed spiking neuron to replace traditional activation functions in ANNs. Unlike conventional binary 0-1 spiking neurons, our ternary design can generate negative spikes as well as 0-1 spikes, providing richer feature information. This allows the network to capture more complex patterns and improves overall detection accuracy.

By incorporating these innovations, our work advances the field of SNN-based object detection and sets a new benchmark for integrating SNNs with established ANN architectures like YOLOX. Extensive experiments validate the effectiveness of our proposed methods, demonstrating significant improvements in detection performance on benchmark datasets.

Related Work

The object detection task involves locating one or multiple objects within an image by drawing bounding boxes around them and identifying their classes. Consequently, deep neural networks (DNN) models for object detection are composed of two primary components: one classifier that identifies the objects, and one regression module that predicts the precise coordinates (x and y axes) and size (width and height) of the bounding boxes. Since accurate prediction of bounding boxes is generally more difficult because of the regression operation, object detection is a significantly more challenging task than image classification.

Yolo Series for Object Detection

Numerous object detection methods have been developed, ranging from DNN-based approaches to Transformer-based models. These methods typically consist of a feature extraction module and a detection head that produces the final results. Among CNN-based models, several high-speed and high-performance methods have emerged, including R-CNN (Girshick et al. 2014), Fast R-CNN (Girshick 2015), Faster R-CNN (Ren et al. 2015), Mask R-CNN (He et al. 2017), SSD (Liu et al. 2016), and the YOLO series (Redmon and Farhadi 2018). Recent iterations of the YOLO series, such as YOLOv8 (Lou et al. 2023), which introduces the C2f building block for enhanced feature extraction and fusion, and YOLOv9 (Wang, Yeh, and Liao 2024), which proposes the GELAN architecture and PGI for augmented training processes, continue to advance the field.

In the realm of Transformer-based object detection, DETR (Carion et al. 2020) stands out as the pioneering model that introduced the Transformer architecture. DETR employs Hungarian loss to achieve one-to-one matching prediction, thereby eliminating the need for hand-crafted components and post-processing steps. Since then, various DETR variants have been proposed to enhance performance and efficiency (Meng et al. 2021; Wang et al. 2022b; Li et al. 2022b; Liu et al. 2022). Deformable-DETR (Zhu et al. 2020) further pushes the boundaries of Transformer-based object detection by leveraging a multi-scale deformable attention module to accelerate convergence.

However, as model parameters continue to increase in both DNN-based and Transformer-based models, there is an urgent need to develop energy-efficient models that strike a balance between performance and computational cost. SNN offers a promising solution by potentially reducing energy consumption while maintaining high performance.

SNN-based Object Detection

Object detection tasks often require SNN with deep architectures to achieve high performance. There are two commonly used approaches for training deep SNNs. The first approach is the ANN-to-SNN conversion method, where ANN are converted into SNNs (Hunsberger and Eliasmith 2015; Rueckauer et al. 2017; Han, Srinivasan, and Roy 2020; Deng and Gu 2021; Simonyan and Zisserman 2014; Li et al. 2021). This approach approximates the average firing rate of SNNs to match the continuous activation values

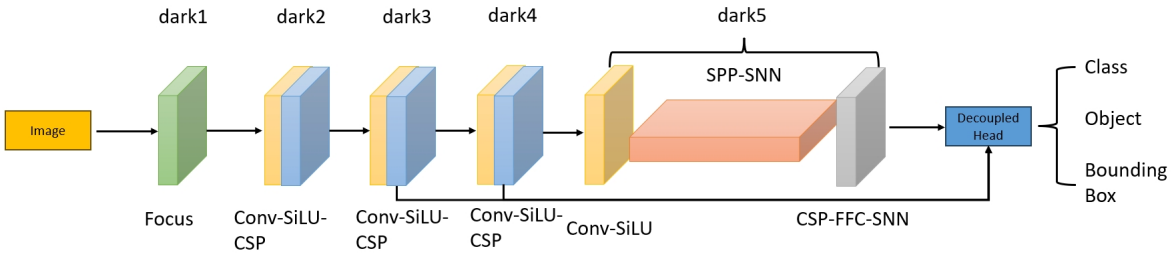

Figure 1: SpikingYOLOX Network. This image shows the simple structure of our SpikingYOLOX.

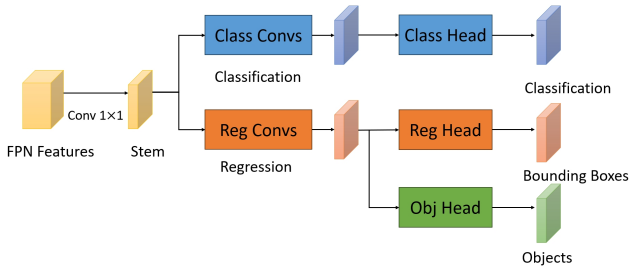

Figure 2: Decoupled Head from YOLOX (Ge et al. 2021). We apply this structure in our SpikingYOLOX.

produced by ReLU nonlinearity in ANNs. Therefore, the performance of the converted SNN is heavily dependent on the accuracy and efficiency of the original ANN. SpikingYOLO (Kim et al. 2020b) represents a pioneering effort to develop a spike-based object detection model. In SpikingYOLO, the introduction of a signed neuron allows for a direct mapping of leaky-ReLU into SNNs. It also implements fine-grained, channel-wise data-based normalization to reduce information loss, though this approach still results in notable performance degradation. Qiu (Qiu et al. 2023) developed a highly efficient and low-latency SNN for object detection, using ANN quantization techniques. The Feed-Forward Integrate-and-Fire (FewdIF) neuron is employed to facilitate high-speed object detection. Furthermore, to support neuromorphic hardware implementations, the region-based SNN (R-SNN) featuring the commonly used IF neuron is proposed in (Jin et al. 2023). The spiking bounding box regressor in R-SNN is designed to decode the spiking trains from output neurons into real-valued bounding box offsets. However, most current ANN-to-SNN models for object detection require a significant number of time steps for the converted SNNs, and their performance remains closely tied to that of the original ANN models.

The second category involves directly training SNNs using surrogate gradients (Neftci, Mostafa, and Zenke 2019). Su introduced EMS-YOLO, an energy-efficient object detection model that utilizes a directly trained SNN with the surrogate gradient method. The EMS-ResNet architecture within EMS-YOLO incorporates fully spike-based residual blocks, enabling low power consumption (Su et al. 2023). Despite these advances, directly trained SNN models for object detection still face challenges, particularly in enhancing the feature extraction capabilities of SNNs to align with the demands of object detection tasks.

Methodology

The goal of our research is to achieve high-performance object detection using SNN. We modify the YOLOX (Ge et al. 2021) backbone for feature extraction and add spiking neurons in the deep layer to mimic how brain works in human beings and to extract features in an adjacent coding. Additionly, the inclusion of FFCs is aimed at leveraging its global receptive field capabilities to enhance feature extraction and improve the overall performance. By combining these innovative elements, our research seeks to push the boundaries of SNN-based object detection and demonstrate its potential for achieving state-of-the-art results. We will introduce our network in details step by step.

Network Structure

In the original YOLOX backbone, the architecture incorporates the DarkNet53 backbone along with Spatial Pyramid Pooling (SPP) layers, similar to YOLOv3-SPP (Bochkovskiy, Wang, and Liao 2020; Jocher 2020). The Cross-Stage Partial Network (CSPNet), initially introduced in YOLOv5 (Jocher 2020), is also utilized in YOLOX for various model sizes (e.g., S, M, L) along with the SiLU activation function. We further modify these modules by integrating spiking neurons for SNN with different model sizes. The backbone of our network is based on the modified CSPNet from YOLOv5. During training, Binary Cross Entropy (BCE) Loss is employed for classification and object branches in the detection head, while Intersection over Union (IoU) Loss is used for the regression branch. These training techniques are orthogonal to the core improvements of YOLOX, and we have maintained them in the training procedure for our SpikingYOLOX network.

In object detection tasks, the conflict between classification and regression (bounding box) is well-known (Song, Liu, and Wang 2020; Wu et al. 2020). YOLOX addresses this by introducing a decoupled detection head, which replaces the traditional coupled head in the YOLO series to enhance convergence speed during end-to-end training. At the beginning of the decoupled head, a 1×1 convolution layer is used to reduce feature channels, followed by two parallel branches, each with two 3×3 convolution layers dedicated

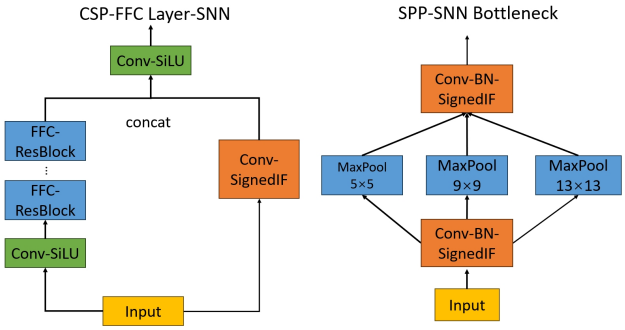

Figure 3: CSP-FFC-SNN (left) and SPP-SNN (right). We use spiking neurons to replace the activation functions in ANN to reduce computation cost and get better gradient for backpropogation.

to classification and regression tasks, respectively. Additionally, an IoU branch is added to the regression branch. Figure 2 illustrates the structure of the decoupled head. We adopt this decoupled head in our models, maintaining the same structure without incorporating spiking neurons.

The overall network structure is depicted in Figure 1. We introduce spiking neurons into the deeper layer of the backbone, precisely, in the dark5 stage, to mimic neuronal processes when working with extracted feature maps to achieve detection results. The CSP layers and the SPP bottleneck feature specially designed for spiking neurons that replace traditional activation functions to enhance performance, which are CSP-FFC-SNN, SPP-SNN, and Signed Spiking Neurons.

CSP-FFC-SNN and SPP-SNN

The Cross Stage Partial (CSP) layer, originally proposed by Wang (Wang et al. 2020) and introduced in object detection by YOLOv4 (Bochkovskiy, Wang, and Liao 2020), sufficiently enhances both detection speed and accuracy. Its partial transition layer is designed to maximize the variation in gradient combinations. In our model, we utilize the fusion-first structure to fully exploit gradient information, as the surrogate function used in spiking neurons requires gradients for backpropagation during training. In the feature extraction branch of the CSP, all convolution layers have been replaced by FFC ResBlocks, which possess a global receptive field for feature extraction. In the shortcut branch, an SNN layer is introduced to replace the pure convolution and activation layers. The SNN's robust feature extraction capabilities against noise allow for precise feature map extraction in complex backgrounds. Both the CSP layer and the SNN branch contribute to reducing computational load, facilitating easier network training. Consequently, in our network, the CSP layer has been transformed into a CSP-FFC-SNN layer, achieving a better balance between high-quality feature maps and computational efficiency. Figure 3 shows the structure of CSP-FFC-SNN. Furthermore, in order to enhance feature map quality, we apply SiLU (Ramachandran, Zoph, and Le 2017), known for its smooth gradient, which aids in effective backpropagation and improves detection performance.

One sigificant advantage of SNNs is their reduced computational cost, owing to their sparse feature coding using discrete spikes rather than the continuous values in ANNs. Spatial Pyramid Pooling (SPP) is widely adopted in object detection for its flexibility and ability to extract high-level feature maps through various pooling layers. However, the primary drawback of SPP is its computational expense during training. The sparse coding of SNNs can significantly mitigate this training cost. Moreover, the fixed value of spikes is ideally suited for pooling layers. Therefore, we replace the activation functions with spiking neurons in the input and output layers, transforming the traditional SPP layer into an SPP-SNN layer. The SPP-SNN structure is also depicted in Figure 3. Note that SPP-SNN layer is positioned in the deepest layer of backbone, serving as a bottleneck layer.

FFC (Chi, Jiang, and Mu 2020) is an operator that leverages global context in early layers and was originally used in image inpainting tasks to generate missing pixels within large masks of high-resolution images. FFC is based on a channel-wise Fast Fourier Transform (FFT) (Brigham and Morrow 1967) , providing a global receptive field that encompasses the entire image. This is crucial for object detection especially when extracting features in deep layers. FFC splits channels into two parallel branches: a local branch with convolution layers only and a global branch using real FFT for global context. Real FFT, applied to real-valued signals, and its inverse ensure that the output remains real-valued. Unlike traditional FFT, which utilizes the entire spectrum, real FFT uses only the positive spectrum, thereby reducing computational cost. FFCs are fully differentiable and can easily replace conventional convolutions, enhancing efficiency by enabling trainable parameters to focus on reasoning and generation rather than merely propagating information.

When applying FFC to our model, we adapt the original FFC to better align with the requirements of object detection and SNN. We change the activation function to ELU (Clevert, Unterthiner, and Hochreiter 2015), which smooths the output in the negative range, avoiding the zero-gradient issue of ReLU and thereby preventing gradient vanishing in deep layers. This adjustment is critical because, in SNN training, extremely deep networks can fail to converge without proper settings. Even a VGG-sized SNN may struggle with convergence if not configured correctly. The properties of ELU and the extensive application of FFC inspire us to incorporate FFC into object detection, leveraging its global receptive field for enhanced feature extraction. FFCs are integrated into FFC residual blocks within the backbone of our network.

Our modified FFC implementation follows these steps:

1. apply Real FFT2d to the input tensor

$$Real\ FFT2d : R^{H\times W\times C} \rightarrow C^{H\times \frac{W}{2}\times C},$$

and concatenate real and imaginary parts

$$ComplexToReal : C^{H\times W\times C} \rightarrow R^{H\times \frac{W}{2}\times 2C};$$

2. apply convolution and activation function in the frequency domain to gain global receptive field

$$ELU \bullet BN \bullet Conv1 \times 1 : R^{H \times \frac{W}{2} \times 2C} \rightarrow R^{H \times \frac{W}{2} \times 2C};$$

3. apply inverse transform to recover a spatial structure

$$RealToComplex : R^{H \times \frac{W}{2} \times 2C} \rightarrow C^{H \times \frac{W}{2} \times C},$$

$$Inverse\ Real\ FFT2d : C^{H \times \frac{W}{2} \times 2C} \rightarrow R^{H \times W \times C}.$$

After processing through the FFC parts, global and local features are fused with a coefficient to balance the ratio of local and global features. In our model, the feature splitting ratio between local and global branches is set at 1:3 (25% local and 75% global). This modified FFC residual block is then applied to our SpikingYOLOX network.

Signed Spiking Neurons

Previous studies (Hu et al. 2023; Wang et al. 2022a) have identified sequential error which is arises when spiking neurons are forced to fire after receiving all potential spikes during the ANN to SNN conversion process. In particular, this conversion loss is eliminated and the firing rates in layer L of the converted SNNs are identical to activations in ANNs:

$$Latency = T \times L = (2^b - 1) \times L, \tag{1}$$

where b and T refers to the conversion bit-precision and conversion time steps. Eq.1 illustrates that imposing a waiting period at each layer could achieve a lossless conversion. this approach becomes impractical when dealing with deep neural networks, particularly those with a large number of layers (L). To address this issue, we propose a novel signed spiking neuron to preserve the integrity of the neural firing pattern without introducing significant latency.

In ANNs, activation functions are employed to introduce non-linearity to model complex relationships within data. In constract, SNNs utilize spiking neurons, which generate binary spikes (0 or 1) as outputs. While this binary representation is sufficient for event-based data, which is inherently binary, it is less suitable for the richer and more complex feature representations required for RGB image data. Simple 0-1 binary spikes are often insufficient to capture the full range of features present in such data. To address this limitation, we propose an enhancement to the spiking neuron model by introducing signed spikes, which allow for a broader and more nuanced range of feature representation. This approach extends the Ignite-and-Fire (IF) neuron model (Izhikevich 2003; Vogels and Abbott 2005; Ponulak and Kasinski 2011), creating a ternary spiking mechanism where the spiking output can take on values of -1, 0, or 1. The key innovation lies in the ability of neurons to generate negative spikes, which provide additional information that enhances the feature extraction process.

Specifically, we impose a condition where neurons are permitted to generate a negative spike only if they have previously generated a positive spike. This is controlled by a secondary threshold, θ', which is set to a small value (1e-3). The spiking function Θ is defined as:

$$\Theta = \begin{cases} 1, & if \quad V_i^l \geq \theta, N = N + 1, \\ -1, & if \quad V_i^l \leq \theta' \quad and \quad N \geq 1, \\ 0, & no\ firing, \end{cases} \tag{2}$$

where N tracks the number of positive spikes a neuron has generated (initialized at 0), and V_i^l is the positive spike threshold. In IF neurons, the total membrane charge $z_i^l(t)$ at time step t is:

$$z_i^l(t) = \sum_{j=1}^{M^{t-1}} W_{ij}^l S_j^{l-1}(t) + b_i^l, \tag{3}$$

where M^{l-1} is the set of neurons at layer $l-1$, W_{ij}^l is the synaptic weight between neurons i and j, b_i^l is the bias term, and $S_j^{l-1}(t)$ indicates an input spike from neuron j at time t. The membrane potential of the IF neuron is updated according to:

$$V_i^l(t) = V_i^l(t-1) + z_i^l(t) - \theta\Theta(V_i^l(t) - \theta), \tag{4}$$

where θ is the firing threshold, and Θ denotes the step function acts as the spike. In our model, Θ is replaces by Eq.2. The membrane potential update equation with the inclusion of negative spikes is given by:

$$\begin{aligned} V_i^l(t) = V_i^l(t-1) + z_i^l(t) - \theta\Theta(V_i^l(t) - \theta) \\ +\theta'\Theta(\theta' - V_i^l(t))\Theta(N-1). \end{aligned} \tag{5}$$

This modified model enables the generation of negative spikes, thereby enhancing the capacity of SNNs to extract features from complex data more effectively. Experimental results demonstrate that networks incorporating these signed spiking neurons outperform those utilizing traditional non-signed IF neurons, indicating the effectiveness of this approach in improving object detection performance.

Experiments

Implementation Details

Our codes are fully implemented in PyTorch, with spiking neurons built by SpikingJelly (Fang et al. 2023). We employ our designed SpikingYOLOX backbone for feature extraction while retaining the same structure of decoupled detection head as used in YOLOX (Ge et al. 2021). We use Adam optimizer (Kingma and Ba 2014) with the step learning rate degradation strategy. The initial learning rate is scale-dependent, with the maximum value of 0.01. All models are trained on the MS-COCO 2017 dataset (Lin et al. 2014), and the total iteration epochs for training varies according to the model scale.

We adopt Mosaic (Bochkovskiy, Wang, and Liao 2020; Jocher 2020) and MixUp (Zhang et al. 2018) as data augmentation techniques. Mosaic helps the network learn bounding box more effectively by exposing more objects within a single image. And MixUp (Zhang et al. 2018), originally developed for image classification and adapted in Bag

Model sizes	mAP(%)	0.5:0.95mAP(%)	0.5mAP(%)	0.75mAP(%)	mAP small(%)	mAP medium(%)	mAP large(%)
Tiny	54.28	34.6	54.0	36.3	12.9	33.9	47.7
S	57.20	37.1	56.7	39.4	16.1	36.5	50.7
M	61.76	41.4	61.4	44.3	20.0	40.9	55.6
L	62.14	42.0	61.6	44.8	21.7	41.6	55.6
X	60.99	41.0	60.6	43.6	19.2	40.7	54.7

Table 1: Quantitative performance of SpikingYOLOX on COCO2017val.

Models	Params(M)	AP val(%)
YOLO-MS-XS	4.5	43.1
YOLOv6-3.0-N	4.7	37.0
YOLOX-tiny	5.1	52.3
SpikingYOLOX-tiny (Ours)	**5.06**	**54.3**
YOLOv6-3.0-S	18.5	44.3
YOLO-MS-S	8.1	46.2
YOLOv8-S	11.2	44.9
YOLOv9-S	7.1	46.7
YOLOX-S	9.0	**57.3**
SpikingYOLOX-S (Ours)	**7.8**	**57.2**

Table 2: Performance comparison on COCO2017val. Results show that in small parameters out SpikingYOLOX could perform the best.

Method	Model	0.5mAP(%)
ANN2SNN	Spiking-YOLO (T=3500)	25.7
	Bayesian Optimization (T=500)	21.1
	Bayesian Optimization (T=5000)	25.8
	Spike Calibration (T=64)	33.1
	Spike Calibration (T=128)	43.6
	Spike Calibration (T=256)	45.3
	Spike Calibration (T=512)	45.4
Directly Trained	EMS-YOLO-ResNet34 (T=4)	50.1
	SpikingYOLOX-Tiny (Ours)	**54.3**
	SpikingYOLOX-S (Ours)	**57.2**

Table 3: Comparison of SNN-based object detection networks on COCO2017val.

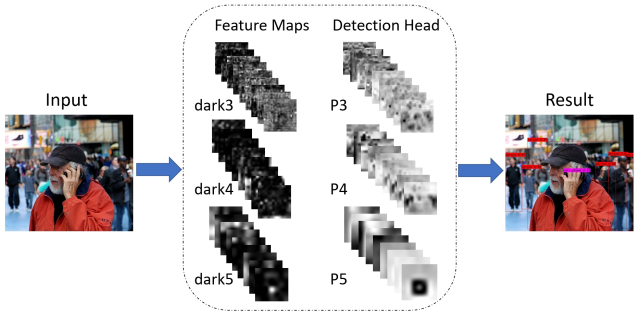

Figure 4: Detection details. In the middle we show the feature maps (left column) and detection results (right column).

Figure 5: Detection results. Different arrows point to the exact output layer of the decoupled head. THe blue arrow points to the IoU result.

of Freebies (BoF) (Zhang et al. 2019) for object detection, blends two images and their corresponding bounding boxes using a coefficient to mitigate overfitting. All model weights are initialized by pretrained YOLOX weights.

Evaluation

We train our network at various scales, following the YOLOX settings, with the exception of the nano size: Tiny, S, M, L, and X. The nano-sized network did not converge during training. Table 2 shows the quantitative results on COCO2017val. It shows that our networks perform well in object detection, especially in average mAP. However, for models with larger parameters, there is still room for improvement in our approach. For instance, the performance of X size is inferior than L size. The reason is mainly due to the huge size that leads to a much slower convergence.

We then compare our SpikingYOLOX with other object detection networks based on the number of parameters and the results are shown in Table 2. It shows that compared to other YOLO series models, our SpikingYOLOX performs better in small parameters, demonstrating the effectiveness of SNNs under these conditions.

We further evaluate our network against other SNN-based object detection methods. Due to the limited number of such studies on MS-COCO, we were able to compare our results with only a few existing methods: Spiking-YOLO (Kim et al. 2020b), Bayesian Optimization (Kim et al. 2020a), Spike Calibration (Li et al. 2022c), and EMS-YOLO (Su et al. 2023). Table 3 shows the results of comparison. The results show that our SpikingYOLOX currently represents the state-of-the-art in SNN-based object detection on the COCO dataset. Remarkably, even the smallest scale of our model surpasses the performance of other SNN-based models, demonstrating the effectiveness and superiority of our approach.

Position	mAP(%)	0.5:0.95mAP(%)	0.5mAP(%)	0.75mAP(%)
Dark5	**57.21**	**37.1**	**56.8**	**39.5**
Dark4	54.50	35.0	54.3	37.1
Dark3	53.95	34.3	53.7	36.3
Dark2	55.60	35.8	55.3	38.2

Table 4: Object detection performance with different spiking neurons positions in backbone.

Neurons	mAP(%)	0.5:0.95mAP(%)	0.5mAP(%)	0.75mAP(%)
Ours	**56.97**	**37.7**	**56.9**	**39.2**
IF Node	56.85	36.8	56.5	39.1

Table 5: Object detection performance with simple IF Node and our tenary signed neurons. The baseline model is the original YOLOX-S.

FFC	mAP(%)	0.5:0.95mAP(%)	0.5mAP(%)	0.75mAP(%)
Base	57.25	**37.7**	57.0	**40.3**
Dark5	**57.67**	37.2	**57.3**	39.3
Dark4&5	54.54	34.9	54.2	37.1

Table 6: Object detection performance with different FFC settings. The baseline is the original YOLOX-S.

Detection Details

We further investigate the procedure of the object detection process, particularly on the quality of the extracted feature maps. Figure 4 illustrates the extracted features within the backbone as well as detection procedures in the detection head. The results demonstrate that our network is capable of extracting meaningful and accurate image features, especially when incorporating signed spiking neurons.

Next, we delve into the details inside the detection head. Having established that our network can generate reasonable feature maps, it is crucial to examine the workings of the detection head to ensure high-performance detection results. Figure 5 shows the details of the detection head. In its deepest layer, we observe 85 channels: 4 channels for bounding boxes, 1 channel for IoU, and 80 channels corresponding to the detection classes of MS-COCO. The detection results, specifically the bounding box predictions, accurately align with the output layer, demonstrating that the feature maps from our SpikingYOLOX backbone are well-suited for high-quality object detection tasks.

Ablation Studies

Spiking Neurons Position

When incorporating spiking neurons to our model, it is vital to determine the optimal placement within the backbone. We experiment by introducing our signed spiking neurons at various points in the backbone. Table 4 shows the performances. For simplicity, we only replaced the activation layers in YOLOX-S with signed spiking neurons, leaving other structures unchanged. The results indicate that the network performs best when the SNNs are applied in the deepest layer. We attribute this to the fact that the loss of features due to spikes, rather than convolution operations, has a significant impact on feature extraction. Therefore, in our SpikingYOLOX model, we position the signed spiking neurons in the dark5 layer of the backbone.

IF-Node and Signed Spiking Neurons

IF node generates only 0-1 spikes, which suffices for tasks like image classification due to its relative simplicity. However, object detection requires the network not only to classify objects but also to precisely locate and bound them, making it a significantly more complex task. Table 5 shows the difference between object detection with IF node and our signed spiking neurons. For a fair comparison, we modified only the activation function to incorporate spiking neurons within the YOLOX-S model. The results indicate a modest but consistent improvement in performance when using our signed spiking neurons. This suggests that the ternary-signed neurons enhance the feature extraction process, making them better suited for the intricate demands of object detection compared to traditional IF neurons.

FFC Settings

A large receptive field is crutial for effcetive feature extraction and FFC with its global receptive field allows networks to reconstruct missing pixels. Thus, in ANN-based object detection networks, stacking multiple convolution layers is the easiest way to achieve a larger receptive field. However, in the realm of signal processing, spectral operations like Fourier transforminherently provide a global receptive field, which is why we have integrated FFC into our SpikingYOLOX model. Table 6 shows our ablation studies about FFC, including two parts: the usage of FFC and the where to best place FFC. The results demonstrate that FFC is most effective when applied in the deepest layer of the backbone. Conversely, excessive stacking of FFC layers can lead to a decline in performance. Therefore, in our SpikingYOLOX model, FFC is strategically utilized only in the dark5 layer, the deepest layer of the backbone.

Conclusions

In this paper, we introduce SpikingYOLOX, which incorporates signed spiking neurons and FFC to enhance feature extraction and overall object detection performance. The signed spiking neurons replace activation functions in the deep layer of backbone, resulting in an SPP-SNN module, while FFC is applied into the CSP-FFC-SNN layer to leverage its global receptive field for improved feature extraction. We train the model across various scales and our experiments demonstrate that these innovative modules are effective across different scales, suggesting their potential adaptability to other computer vision tasks. Extensive experiments show that SpikingYOLOX achieves outstanding performance in the domain of SNN-based object detection networks, clearly illustrating its advantages and potential as a high-performance SNN architecture in many other tasks.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant (No. 62476035, 62206037, U24B20140 and 62306274), and the China Scholarship Council (CSC) grants, and the Open Research Fund of the State Key Laboratory of Brain-Machine Intelligence, Zhejiang University (Grant No. BMI2400012).

References

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Brigham, E. O.; and Morrow, R. 1967. The fast Fourier transform. *IEEE spectrum*, 4(12): 63–70.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

Chi, L.; Jiang, B.; and Mu, Y. 2020. Fast fourier convolution. *Advances in Neural Information Processing Systems*, 33: 4479–4488.

Clevert, D.-A.; Unterthiner, T.; and Hochreiter, S. 2015. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*.

Deng, S.; and Gu, S. 2021. Optimal conversion of conventional artificial neural networks to spiking neural networks. *arXiv preprint arXiv:2103.00476*.

Fang, W.; Chen, Y.; Ding, J.; Yu, Z.; Masquelier, T.; Chen, D.; Huang, L.; Zhou, H.; Li, G.; and Tian, Y. 2023. Spikingjelly: An open-source machine learning infrastructure platform for spike-based intelligence. *Science Advances*, 9(40): eadi1480.

Ge, Z.; Liu, S.; Wang, F.; Li, Z.; and Sun, J. 2021. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*.

Girshick, R. 2015. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 1440–1448.

Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.

Han, B.; Srinivasan, G.; and Roy, K. 2020. Rmp-snn: Residual membrane potential neuron for enabling deeper high-accuracy and low-latency spiking neural network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13558–13567.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

Hu, Y.; Tang, H.; and Pan, G. 2021. Spiking deep residual networks. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8): 5200–5205.

Hu, Y.; Zheng, Q.; Jiang, X.; and Pan, G. 2023. Fast-snn: Fast spiking neural network by converting quantized ann. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Hunsberger, E.; and Eliasmith, C. 2015. Spiking deep networks with LIF neurons. *arXiv preprint arXiv:1510.08829*.

Hwang, S.; Chang, J.; Oh, M.-H.; Min, K. K.; Jang, T.; Park, K.; Yu, J.; Lee, J.-H.; and Park, B.-G. 2021. Low-latency spiking neural networks using pre-charged membrane potential and delayed evaluation. *Frontiers in Neuroscience*, 15: 629000.

Izhikevich, E. M. 2003. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6): 1569–1572.

Jin, X.; Zhang, M.; Yan, R.; Pan, G.; and Ma, D. 2023. R-snn: Region-based spiking neural network for object detection. *IEEE Transactions on Cognitive and Developmental Systems*.

Jocher, G. 2020. Ultralytics YOLOv5.

Kim, S.; Park, S.; Na, B.; Kim, J.; and Yoon, S. 2020a. Towards fast and accurate object detection in bio-inspired spiking neural networks through Bayesian optimization. *IEEE Access*, 9: 2633–2643.

Kim, S.; Park, S.; Na, B.; and Yoon, S. 2020b. Spiking-yolo: spiking neural network for energy-efficient object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 11270–11277.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. 2022a. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*.

Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L. M.; and Zhang, L. 2022b. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13619–13627.

Li, Y.; Deng, S.; Dong, X.; Gong, R.; and Gu, S. 2021. A free lunch from ANN: Towards efficient, accurate spiking neural networks calibration. In *International conference on machine learning*, 6316–6325. PMLR.

Li, Y.; He, X.; Dong, Y.; Kong, Q.; and Zeng, Y. 2022c. Spike calibration: Fast and accurate conversion of spiking neural network for object detection and segmentation. *arXiv preprint arXiv:2207.02702*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Liu, S.; Li, F.; Zhang, H.; Yang, X.; Qi, X.; Su, H.; Zhu, J.; and Zhang, L. 2022. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, 21–37. Springer.

Lou, H.; Duan, X.; Guo, J.; Liu, H.; Gu, J.; Bi, L.; and Chen, H. 2023. DC-YOLOv8: small-size object detection algorithm based on camera sensor. *Electronics*, 12(10): 2323.

Meng, D.; Chen, X.; Fan, Z.; Zeng, G.; Li, H.; Yuan, Y.; Sun, L.; and Wang, J. 2021. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3651–3660.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6): 51–63.

Ponulak, F.; and Kasinski, A. 2011. Introduction to spiking neural networks: Information processing, learning and applications. *Acta neurobiologiae experimentalis*, 71(4): 409–433.

Qiu, N.; Li, Z.; Li, Y.; and Zhu, C. 2023. Highly Efficient SNNs for High-speed Object Detection. *arXiv preprint arXiv:2309.15883*.

Ramachandran, P.; Zoph, B.; and Le, Q. V. 2017. Searching for activation functions. *arXiv preprint arXiv:1710.05941*.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Redmon, J.; and Farhadi, A. 2017. YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7263–7271.

Redmon, J.; and Farhadi, A. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Rueckauer, B.; Lungu, I.-A.; Hu, Y.; Pfeiffer, M.; and Liu, S.-C. 2017. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in neuroscience*, 11: 294078.

Samadzadeh, A.; Far, F. S. T.; Javadi, A.; Nickabadi, A.; and Chehreghani, M. H. 2023. Convolutional spiking neural networks for spatio-temporal feature extraction. *Neural Processing Letters*, 55(6): 6979–6995.

Shen, J.; Xu, Q.; Liu, J. K.; Wang, Y.; Pan, G.; and Tang, H. 2023. Esl-snns: An evolutionary structure learning strategy for spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 86–93.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Song, G.; Liu, Y.; and Wang, X. 2020. Revisiting the sibling head in object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11563–11572.

Su, Q.; Chou, Y.; Hu, Y.; Li, J.; Mei, S.; Zhang, Z.; and Li, G. 2023. Deep directly-trained spiking neural networks for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6555–6565.

Tavanaei, A.; Ghodrati, M.; Kheradpisheh, S. R.; Masquelier, T.; and Maida, A. 2019. Deep learning in spiking neural networks. *Neural networks*, 111: 47–63.

Vogels, T. P.; and Abbott, L. F. 2005. Signal propagation and logic gating in networks of integrate-and-fire neurons. *Journal of neuroscience*, 25(46): 10786–10795.

Wang, C.-Y.; Bochkovskiy, A.; and Liao, H.-Y. M. 2023. 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*, 7464–7475.

Wang, C.-Y.; Liao, H.-Y. M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; and Yeh, I.-H. 2020. CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 390–391.

Wang, C.-Y.; Yeh, I.-H.; and Liao, H.-Y. M. 2024. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616*.

Wang, Y.; Zhang, M.; Chen, Y.; and Qu, H. 2022a. Signed Neuron with Memory: Towards Simple, Accurate and High-Efficient ANN-SNN Conversion. In *IJCAI*, 2501–2508.

Wang, Y.; Zhang, X.; Yang, T.; and Sun, J. 2022b. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 2567–2575.

Wu, Y.; Chen, Y.; Yuan, L.; Liu, Z.; Wang, L.; Li, H.; and Fu, Y. 2020. Rethinking classification and localization for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10186–10195.

Xu, Q.; Li, Y.; Shen, J.; Liu, J. K.; Tang, H.; and Pan, G. 2023. Constructing deep spiking neural networks from artificial neural networks with knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7886–7895.

Xu, Q.; Li, Y.; Shen, J.; Zhang, P.; Liu, J. K.; Tang, H.; and Pan, G. 2022. Hierarchical spiking-based model for efficient image classification with enhanced feature extraction and encoding. *IEEE Transactions on Neural Networks and Learning Systems*.

Xu, Q.; Qi, Y.; Yu, H.; Shen, J.; Tang, H.; Pan, G.; et al. 2018. Csnn: an augmented spiking based framework with perceptron-inception. In *IJCAI*, volume 1646.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*.

Zhang, Z.; He, T.; Zhang, H.; Zhang, Z.; Xie, J.; and Li, M. 2019. Bag of freebies for training object detection neural networks. *arXiv preprint arXiv:1902.04103*.

Zheng, H.; Wu, Y.; Deng, L.; Hu, Y.; and Li, G. 2021. Going deeper with directly-trained larger spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 11062–11070.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2020. Deformable detr: Deformable transformers for end-to-end object detection. arXiv 2020. *arXiv preprint arXiv:2010.04159*.