

SpikeHCD: Spiking Transformer With Parallel Neurons and Memory-Enhanced Attention for Hyperspectral Change Detection

Zihao Mei^{ID}, Jianhao Li^{ID}, Bolin Zhang^{ID}, Chong Wang^{ID}, Member, IEEE, Lijun Guo^{ID}, Member, IEEE, and Jiangbo Qian^{ID}, Member, IEEE

Abstract—The hyperspectral change detection (CD) is a critical technology in remote sensing, widely applied in urban planning, environmental monitoring, and disaster detection. However, the hyperspectral data exhibits higher spectral dimensionality compared to conventional RGB data, making existing methods struggle to balance high accuracy and low energy consumption. As the third generation of neural networks, spiking neural networks (SNNs) demonstrate the advantage of low energy efficiency, but the iterative computational process in spiking neurons significantly increases training and inference burdens when applied to hyperspectral CD. To address these challenges, we propose a novel spiking Transformer with parallel neurons and memory-enhanced attention for hyperspectral CD named SpikeHCD, the first SNNs specifically designed for hyperspectral CD. SpikeHCD not only maintains a low-energy advantage but also employs a probability-driven parallel spiking neurons (PPSN) to improve the computational efficiency, enabling more effective application in remote sensing tasks. We further design a memory-enhanced spiking attention (MSA) module to enhance temporal modeling capability, and thoroughly extract spatial-spectral features. In addition, a spiking difference module (SDM) is introduced to capture change features across different timesteps. Experimental results demonstrate that SpikeHCD can achieve several state-of-the-art (SOTA) results on multiple hyperspectral datasets, with faster detection, lower energy consumption, and fewer number of parameters. The codes are available at: https://github.com/mzhcode/HCD_snn

Index Terms—Attention mechanism, change detection (CD), hyperspectral image (HSI), spiking neural networks (SNNs).

I. INTRODUCTION

HYPERSPECTRAL change detection (CD) leverages rich spectral information to analyze dynamic variations between multitemporal hyperspectral images (HSIs) of the

Received 1 May 2025; revised 28 August 2025 and 2 November 2025; accepted 17 November 2025. Date of publication 25 November 2025; date of current version 4 December 2025. This work was supported in part by Zhejiang Provincial Natural Science Foundation of China under Grant Z26D010010 and Grant ZCLQN25F0207; in part by NSF, China, under Grant 62271274; in part by Ningbo S&T Project under Grant 2024Z004 and Grant 2023Z059; and in part by the K. C. Wong Magna Fund in Ningbo University. (Corresponding author: Jiangbo Qian.)

Zihao Mei, Jianhao Li, Bolin Zhang, and Jiangbo Qian are with the Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China (e-mail: 2211100269@nbu.edu.cn; ljh57@foxmail.com; zhangbolin@nbu.edu.cn; qianjiangbo@nbu.edu.cn).

Chong Wang and Lijun Guo are with the Merchants' Guild Economics and Cultural Intelligent Computing Laboratory, Ningbo University, Ningbo 315211, China (e-mail: wangchong@nbu.edu.cn; guolijun@nbu.edu.cn).

Digital Object Identifier 10.1109/TGRS.2025.3636865

same geographic region, enabling large-scale monitoring of land surface changes. However, compared with the conventional RGB data, hyperspectral data exhibits significantly higher spectral dimensionality. When deploying existing artificial neural network-based methods on resource-constrained satellite devices, it is particularly challenging to simultaneously maintain both accuracy and energy efficiency. There is an urgent demand to develop an energy-efficient and high-performance architecture for hyperspectral CD. As the third generation of neural networks, spiking neural networks (SNNs) encode information through binary spikes and have demonstrated great potential in promoting energy efficiency of hyperspectral CD. Consequently, for the first time, we propose a novel spiking Transformer with parallel neurons and memory-enhanced attention for hyperspectral CD named SpikeHCD.

Although applying SNNs to hyperspectral CD can significantly improve energy efficiency, existing methods have to face two critical challenges.

- 1) *Accuracy Limitations*: Temporal information and change features are important in hyperspectral CD, as land surface changes often exhibit periodic dynamic patterns, such as seasonal cycles of farmland, and the growth and withering of vegetation. These temporal features provide rich contextual relationships, enabling models to better understand land-cover evolution and change trends. However, current SNN methods primarily focus on spatial feature extraction and difficult to capture the complex change features and temporal dependencies in multitemporal hyperspectral data, limiting their performance in CD tasks.
- 2) *Computational Efficiency*: The iterative computational process in spiking neurons inevitably increases the temporal overhead during both training and inference. In remote sensing applications, particularly when processing large-scale hyperspectral datasets, improving computational efficiency is essential.

To address these challenges, SpikeHCD incorporates three innovative modules for optimization. First, to enhance detection accuracy, we propose an iterative memory-enhanced spiking attention (MSA), which enhances temporal modeling capabilities by incorporating historical information into attention interactions. Notably, MSA is implemented in a

parallel way with no additional temporal overhead. Furthermore, we propose a spiking difference module (SDM) to model difference features across timesteps, enabling efficient extraction of change features from multitemporal hyperspectral data. Second, to improve computational efficiency, we design a probability-driven parallel spiking neuron (PPSN) that estimates spike firing probabilities through the extrema of membrane potential and replaces the traditional spike-driven reset mechanism with a probability-driven reset mechanism. PPSN achieves parallelized spike computation that significantly accelerates both training and inference. The main contributions of this article are as follows.

- 1) We propose a PPSN that replaces the conventional spike-driven reset mechanism with a novel probability-driven reset mechanism, enabling parallelization of the iterative spiking computational process. PPSN significantly enhances both training and inference efficiency, making SNNs more efficient for remote sensing applications.
- 2) We propose an iterative MSA mechanism that incorporates historical information into attention interactions, effectively capturing temporal dynamics in multitemporal HSIs. MSA is implemented in a parallel way, with no additional temporal overhead.
- 3) We propose a novel spiking Transformer with parallel neurons and memory-enhanced attention for hyperspectral CD named SpikeHCD. By integrating MSA and PPSN modules, SpikeHCD effectively extracts spatiotemporal-spectral features from multitemporal hyperspectral data.

The rest of this article is structured as follows. Section II provides a brief review of related work. Section III presents a detailed description of our proposed SpikeHCD. Section IV validates the effectiveness of SpikeHCD through extensive experiments. Section V concludes our work.

II. RELATED WORKS

A. Hyperspectral Change Detection

During the early years, traditional methods [1], [2], [3], [4], [5], [6], [7], [8] were widely used to monitor land cover changes, including algebra-based methods, post-classification methods, transformation-based approaches, and so on. With the rapid growth of deep learning, there has been a significant increase in deep learning-based hyperspectral CD methods [9], [10], [11], [12]. Wang et al. [13] proposed an end-to-end CNN network (SiamNet) incorporating spectral-spatial attention mechanisms, which enhances important spectral channels and spatial locations while suppressing less informative features. Yang et al. [14] proposed a deep multiscale pyramid network with spatial-spectral residual attention mechanism to improve the detection performance in complex change regions. Shi et al. considered the correlation between spatial and spectral information in hyperspectral data and employed a specifically designed 3-D convolutional neural network to extract features from multitemporal HSIs along both spatial and spectral dimensions [15].

Recent studies have also introduced Transformer into hyperspectral CD. Ding et al. [16] introduced the CDFFormer,

which utilizes spatial-temporal position encoding and multihead attention mechanisms to fuse spatiotemporal-spectral features. Song et al. [17] developed the cross-temporal interaction symmetric attention network (CSANet), achieving the feature interaction between bitemporal hyperspectral data through symmetric attention modules. Wang et al. [18] further improved the 3-D Transformer architecture by proposing SSTFormer, which implements multidimensional feature extraction through a spectral-spatial-temporal network. SSTFormer introduced class tokens to store categorical information, combined with spatial encoding and temporal difference encoding, demonstrating superior performance in complex land-cover CD. To address the limitations of Transformer in local feature modeling, Wang et al. [19] proposed a global and local attention-based Transformer (GLAFormer), where the global attention branch captures low-frequency information through downsampled features, while the local attention branch focuses on high-frequency details using nonoverlapping window-based local attention. Ding et al. [20] constructed a multilevel feature fused and change information enhanced neural network (MFCEN) that guides high-level semantic features to focus on change regions using low-level spatial details, while high-level semantic features reciprocally enhance the discriminative capability of low-level features, enabling the learned features to contain richer semantic information and capture intrinsic correlations between different features. MsFNet [21] incorporates a dynamic convolution module with spectral attention to adaptively modulate receptive fields and enhance critical band focus, alongside a multiscale feature fusion module to enrich local information. BTCDNet [22] integrates a Bayesian prior guided module (BPGM) to improve robustness under limited labeled samples and a tile attention block (TAB) to reduce complexity and elevate performance. Meanwhile, AIWSEN [23] employs an adaptive information weighting attention module (AIWAM) based on the information entropy to capture differential features, a dual-time synchronic change enhancing module (DSCEM) to extract interactive features from bitemporal images, and a bitemporal feature selection and fusion module (BFSFM) with gating mechanisms to filter and fuse features. Although these models have achieved performance improvements through various methods, the higher spectral dimensionality of hyperspectral data compared with the conventional data makes it challenging for these methods to simultaneously maintain the requirements of both high accuracy and low energy consumption.

B. Spiking Neural Network

Transformers have demonstrated outstanding performance in numerous fields, such as natural language processing (NLP) and computer vision (CV), attracting significant attention and research [24], [25]. In recent years, researchers have explored various Transformer architectures adapted to the characteristics of SNNs. Zhou et al. [26] first proposed a Transformer architecture based on SNNs named Spikformer, which employs an attention mechanism specifically designed for SNNs, using spike-form Q , K , and V to compute the attention matrix. Yao et al. [27], [28] proposed the spike-driven Transformer, which transforms matrix multiplication in

the self-attention operator into Hadamard product-based masking operations and replaces softmax and scaling operations with matrix column summation, drastically reducing energy consumption and simplifying the model. Zhou et al. [29], [30] proposed Spikingformer, which further avoids non-spiking computations in Spikformer while maintaining high model performance. Shi et al. [31] proposed a hierarchical SpikingResformer to address the current limitations of SNN-Transformers in local feature extraction. Zhou et al. [32] proposed a hierarchical spiking Transformer framework named QKFormer, where the Q-K attention contains only \mathbf{Q} and \mathbf{K} vectors and uses binary spike vectors to model feature dimension importance, significantly reducing model complexity. Fang et al. proposed a parallel spiking neuron (PSN) [33] that achieves parallel computation by eliminating the membrane potential reset mechanism in spiking neurons, providing a new paradigm for efficient training of deep SNNs. In current SNN research, attention mechanisms are often used to enhance the focus on the temporal dimension by evaluating the importance of different timesteps. Wang et al. [34] proposed the STS-Transformer, introducing a spatial-temporal self-attention mechanism that preserves the asynchronous characteristics of SNNs, along with spatial-temporal relative position bias (STRPB). Yao et al. [35] proposed TA-SNN, which incorporates a temporal attention mechanism to enable the model to effectively learn frame-level representations of event streams and determine frame importance during training while discarding irrelevant frames during inference. Zhu et al. [36] proposed TCJA-SNN, which uses two local attention mechanisms for temporal and channel feature extraction and employs a cross-convolution fusion (CCF) layer to model dependencies between temporal and channel features, further enhancing model performance. Yao et al. [37] proposed attention SNN, which introduces a multidimensional attention module to focus on important information.

C. Summary

The high spectral dimensionality of hyperspectral data presents a significant challenge in simultaneously maintaining both high precision and low energy consumption requirements. While SNNs demonstrate remarkable energy efficiency advantages, their application to hyperspectral CD tasks remains constrained by accuracy and computational efficiency limitations. In this article, we propose SpikeHCD, the first SNNs specifically designed for hyperspectral CD.

III. METHODS

We propose a novel SNN specifically designed for hyperspectral CD, named SpikeHCD. SpikeHCD incorporates three novel modules: MSA module, SDM, and PPSN. We first outline the basic principles of spiking neurons and PPSN, followed by a detailed introduction to the overall architecture of SpikeHCD.

A. Parallel Spiking Neuron Layer

1) *Traditional Spiking Neuron:* Spiking neuron is the essential core component in SNNs. Over the years, various

spiking neuron models have been proposed, including the leaky integrate-and-fire (LIF) model [38], the Hodgkin-Huxley (H-H) model [39], and the Izhikevich model [40]. The LIF neuron model is the most widely used model and can simulate complex dynamic processes of biological neurons while maintaining a simplified mathematical representation. The LIF neuron model iterates over multiple time steps, with its internal operations divided into three processes, which can be expressed with the following equations:

$$H[t] = V[t-1] + \frac{1}{\gamma} (X[t] - (V[t-1] - V_{\text{reset}})) \quad (1)$$

$$S[t] = \text{Hea}(H[t] - v_{\text{th}}) \quad (2)$$

$$V[t] = V_{\text{reset}} S[t] + H[t] (1 - S[t]). \quad (3)$$

Equation (1) represents the charging process, where $H[t]$ is the membrane potential generated by combining spatial input from the current time step $X[t]$ with the reset membrane potential from the previous time step $V[t-1]$. The membrane time constant γ attenuates the input. Equation (2) represents the firing process, where $\text{Hea}(\cdot)$ denotes the Heaviside step function. When the membrane potential $H[t]$ exceeds the threshold v_{th} , $S[t] = 1$, indicating that the spiking neuron fires a spike. Conversely, $S[t] = 0$, indicating that no spike is fired. Equation (3) represents the resetting process, where $V[t]$ is the membrane potential after resetting at the current time step. If a spike is fired, $V[t]$ is set to the reset potential V_{reset} ; otherwise, it retains the original membrane potential $H[t]$. Typically, V_{reset} is set to 0 in Section III-A2.

2) *Probability-Driven Parallel Spiking Neuron:* As shown in Fig. 1(a), the vanilla LIF neuron [38] must perform computations iteratively along the temporal dimension due to the reset process (3). When applying SNNs to hyperspectral CD tasks, this iterative computational process inevitably introduces significant temporal overhead. Fig. 1(b) illustrates that the existing PSN method [33] achieves parallel computation by removing the reset process. Although this method improves computational efficiency, such simplification adversely affects the spatiotemporal dynamics of spiking neurons, making it difficult to achieve high performance. To address these challenges, we propose a novel PPSN. PPSN can be divided into two steps: 1) it estimates the spike firing probability at each timestep by calculating the extrema of membrane potential and 2) it replaces the traditional spike-driven reset mechanism with a probability-driven membrane potential reset mechanism (P-D reset), thereby achieving parallelization while preserving the spatiotemporal dynamics of spiking neurons.

First, we calculate the extrema (maximum and minimum values) of membrane potential at each timestep.

At timestep t , the maximum value of membrane potential $H[t]_{\max}$ is defined as follows. When no spike is fired in all previous $t-1$ timesteps, the complete historical membrane potential propagates to the current timestep t , resulting in the maximum membrane potential value $H[t]_{\max}$. The formula is as follows:

$$H[t]_{\max} = \frac{1}{\gamma} \sum_{k=0}^t \left(1 - \frac{1}{\gamma}\right)^{t-k} X[k]. \quad (4)$$

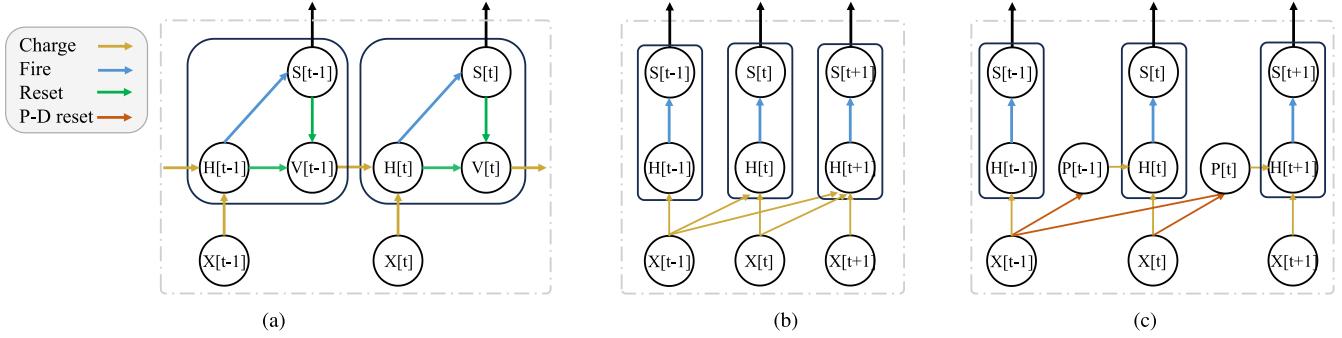


Fig. 1. Illustration of three types of spiking neurons. (a) Vanilla spiking neuron. (b) PSN spiking neuron. (c) Ours spiking neuron PPSN.

At timestep t , the minimum value of membrane potential $H[t]_{\min}$ is defined as follows. When a spike is fired at the preceding timestep $t - 1$, the membrane potential propagated to the current timestep t becomes 0, resulting in the minimum membrane potential value $H[t]_{\min}$. The formula is as follows:

$$H[t]_{\min} = \frac{1}{\gamma} X[t]. \quad (5)$$

Second, we design a probability-driven membrane potential reset mechanism (P-D reset) based on the extrema of membrane potential. Specifically, we use the mean value of the maximum value $H[t]_{\max}$ and minimum value $H[t]_{\min}$ of membrane potential as the estimated membrane potential, and calculate the spike firing probability $P[t]$ through a sigmoid function. The membrane potential is then reset based on this spike firing probability $P[t]$. The formulas are as follows:

$$P[t] = \text{Sigmoid}\left(\frac{1}{2}(H[t]_{\max} + H[t]_{\min}) - v_{\text{th}}\right) \quad (6)$$

$$V[t] = H[t](1 - P[t]). \quad (7)$$

By replacing the traditional spike-driven reset mechanism with P-D reset [substituting (3) into (7)], we can reformulate the iterative computation of spiking neurons into parallelizable matrix multiplications.

Taking the total number of timesteps $T = 4$ as an example. First, we calculate the extrema of the membrane potential. The derivation can be expressed as follows: $H[0]_{\max} = 1/\gamma X[0]$, $H[1]_{\max} = 1/\gamma(\alpha X[0] + X[1])$, $H[2]_{\max} = 1/\gamma(\alpha^2 X[0] + \alpha X[1] + X[2])$, and $H[3]_{\max} = 1/\gamma(\alpha^3 X[0] + \alpha^2 X[1] + \alpha X[2] + X[3])$, where $\alpha = 1 - 1/\gamma$. Based on this derivation, we can construct a coefficient matrix $A \in \mathbb{R}^{T \times T}$. The maximum potential $H_{\max} = \{H[0]_{\max}, H[1]_{\max}, H[2]_{\max}, H[3]_{\max}\}$ can be obtained through matrix multiplication between A and input X . Subsequently, the probability matrix P can be calculated using H_{\max} and H_{\min} . The formulas are as follows:

$$H_{\max} = AX \quad (8)$$

$$A = \frac{1}{\gamma} \begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha & 1 & 0 & 0 \\ \alpha^2 & \alpha & 1 & 0 \\ \alpha^3 & \alpha^2 & \alpha & 1 \end{bmatrix} \quad (9)$$

$$H_{\min} = \frac{1}{\gamma} X \quad (10)$$

$$P = \text{Sigmoid}\left(\frac{1}{2}(H_{\max} + H_{\min}) - v_{\text{th}}\right). \quad (11)$$

After obtaining the probability matrix P , we substitute (7) into (1), where $E = 1 - P$, and eliminate V through algebraic elimination, yielding the following simplified formulation:

$$H[t] = \alpha H[t-1] E[t-1] + \frac{1}{\gamma} X[t]. \quad (12)$$

Based on (12), the membrane potential at each timestep can be expressed as: $H[0] = 1/\gamma X[0]$, $H[1] = 1/\gamma(\alpha E[0]X[0] + X[1])$, $H[2] = 1/\gamma(\alpha^2 E[0]E[1]X[0] + \alpha E[1]X[1] + X[2])$, and $H[3] = 1/\gamma(\alpha^3 E[0]E[1]E[2]X[0] + \alpha^2 E[1]E[2]X[1] + \alpha E[2]X[2] + X[3])$. Following this derivation, we can construct a coefficient matrix $B \in \mathbb{R}^{T \times T}$, where the membrane potential H can be obtained through matrix multiplication between B and input X , ultimately computing the spike sequence S . The formulas are as follows:

$$H = BX \quad (13)$$

$$B = \frac{1}{\gamma} \begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha E[0] & 1 & 0 & 0 \\ \alpha^2 E[0] E[1] & \alpha E[1] & 1 & 0 \\ \alpha^3 E[0] E[1] E[2] & \alpha^2 E[1] E[2] & \alpha E[2] & 1 \end{bmatrix} \quad (14)$$

$$S = \text{Hea}(H - v_{\text{th}}). \quad (15)$$

The iterative computational process of spiking neurons can be reformulated into a parallelized form through the computational procedure, which significantly improves the computational efficiency of SNNs and enables more efficient applications for remote sensing tasks.

B. Overall Architecture

The overall architecture of SpikeHCD is shown in Fig. 2. The multitemporal HSI input is represented as $X_1, X_2 \in \mathbb{R}^{C \times H \times W}$, where C represents the spectral dimension, and H and W are the height and width, respectively. Before input to the network, each image is duplicated to construct a four-timestep image sequence, which is then divided into $p \times p$ patches, finally obtaining the input $X \in \mathbb{R}^{4 \times C \times p \times p}$. SpikeHCD consists of four modules: spiking patch embedding (SPE) module, memory-enhanced spiking Transformer block, SDM, and classification layer. First, X is mapped to appropriate dimensions through the SPE module. The features are then input to the memory-enhanced spiking

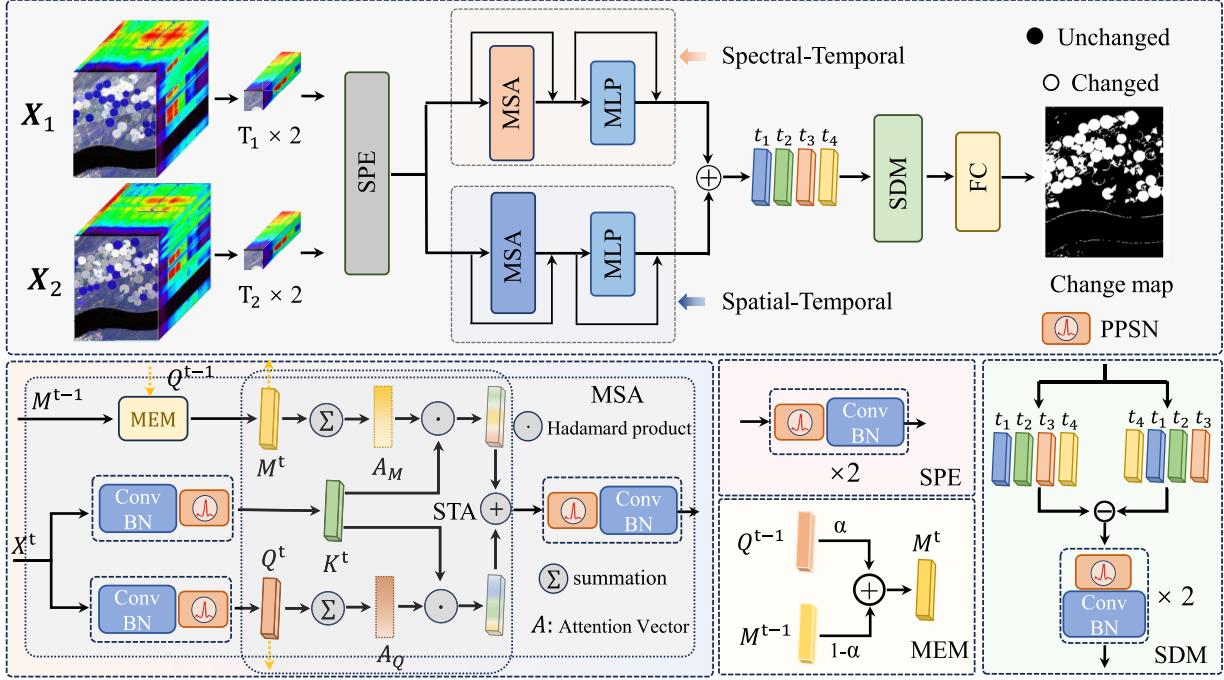


Fig. 2. Architecture of SpikeHCD is comprised of: an SPE module, a dual-branch memory-enhanced spiking Transformer block incorporating MSA and MLP modules, an SDM, and a classification layer [fully connected layer (FC)].

Transformer block to extract spatiotemporal-spectral features. The memory-enhanced spiking Transformer block has a dual-branch structure, with each branch composed of a MSA and a multilayer perceptron (MLP). The modules in the upper branch focus on spectral feature fusion, while those in the lower branch mainly extract spatial features. The features output from the two branches are fused through addition and input to the SDM to model the difference features. Finally, SpikeHCD outputs the CD results for each pixel through the classification layer. Throughout the entire network architecture, we employ our proposed PPSN to improve the training and inference efficiency.

C. Memory-Enhanced Spiking Transformer Block

The overall architecture of the memory-enhanced spiking Transformer block is illustrated in Fig. 2, employing a dual-branch design where each branch consists of our proposed MSA and MLP. Conventional spiking self-attention (SSA) mechanisms are limited to feature interactions within individual timesteps, making it difficult to effectively capture temporal dependencies across multitemporal HSIs. To address this challenge, MSA is proposed. MSA performs attention computation across multiple timesteps while retaining historical information \mathbf{M} from previous timesteps for feature interaction, thereby enhancing the temporal modeling capability. Notably, MSA is implemented in a parallel way during both training and inference without introducing additional temporal overhead. The formulas for obtaining \mathbf{Q} , \mathbf{K} , and \mathbf{M} are as follows:

$$\mathbf{Q} = \text{PPSN}(\text{BN}(\text{Conv}(\mathbf{X}))) \quad (16)$$

$$\mathbf{K} = \text{PPSN}(\text{BN}(\text{Conv}(\mathbf{X}))) \quad (17)$$

$$\mathbf{M}^t = \lambda \cdot \mathbf{Q}^{t-1} + (1 - \lambda) \cdot \mathbf{M}^{t-1}. \quad (18)$$

Here, $\mathbf{X} \in \mathbb{R}^{T \times N \times D}$ represents the input spiking sequences, $\text{Conv}(\cdot)$ denotes a convolutional layer with a kernel size of 1×1 , $\text{BN}(\cdot)$ indicates batch normalization, and $\text{PPSN}(\cdot)$ is our proposed PSN. \mathbf{M}^t denotes the historical information features at the timestep t , obtained through weighted summation of previous historical features \mathbf{M}^{t-1} and query \mathbf{Q}^{t-1} , where $\lambda \in [0, 1]$ denotes a learnable historical information decay parameter.

MSA is implemented in a parallel way with no additional temporal overhead. While MSA captures temporal information through iterative computation and feature interaction of historical information \mathbf{M} , the calculation process of \mathbf{M} can be efficiently implemented via parallelizable matrix multiplications. Taking the total number of timesteps $T = 4$ as an example: $\mathbf{M}^0 = 0$, $\mathbf{M}^1 = \lambda \cdot \mathbf{Q}^0$, $\mathbf{M}^2 = \lambda \cdot \mathbf{Q}^1 + (1 - \lambda) \cdot \lambda \cdot \mathbf{Q}^0$, and $\mathbf{M}^3 = \lambda \cdot \mathbf{Q}^2 + \lambda \cdot (1 - \lambda) \cdot \mathbf{Q}^1 + (1 - \lambda)^2 \cdot \lambda \cdot \mathbf{Q}^0$. Based on this recursive relationship, we construct a coefficient matrix $\mathbf{Z} \in \mathbb{R}^{T \times T}$ and compute \mathbf{M} through matrix multiplication between \mathbf{Q} and \mathbf{Z} . The formulas are as follows:

$$\mathbf{Z} = \lambda \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ (1 - \lambda) & 1 & 0 & 0 \\ (1 - \lambda)^2 & (1 - \lambda) & 1 & 0 \end{bmatrix} \quad (19)$$

$$\mathbf{M} = \mathbf{ZQ}. \quad (20)$$

After obtaining the spike-form \mathbf{Q} , $\mathbf{K} \in \mathbb{R}^{T \times N \times D}$ and historical information representation $\mathbf{M} \in \mathbb{R}^{T \times N \times D}$, the subsequent computational process of MSA is formulated as follows:

$$\mathbf{X} = \text{STA}(\mathbf{Q}, \mathbf{K}, \mathbf{M}) \quad (21)$$

$$\mathbf{X} = \text{Linear}(\text{PPSN}(\mathbf{X})). \quad (22)$$

$\text{STA}(\cdot)$ represents the attention computational process and $\text{Linear}(\cdot)$ denotes a linear layer. Based on the distinction of feature fusion, and $\text{STA}(\cdot)$ can be classified into spatial-temporal

attention module (STA1) and spectral–temporal attention module (STA2).

STA1 focuses on extracting spatial–temporal features. Its core operation involves compressing \mathbf{Q} and \mathbf{M} into vectors to generate a weight value for each spatial position. These weights are then multiplied with \mathbf{K} to enhance responses at important spatial locations while suppressing irrelevant regions. The formulas are as follows:

$$\text{STA1} : \begin{cases} \mathbf{A}_Q^{\text{Spatial}} = \sum_{i=0}^D \mathbf{Q}_i \\ \mathbf{A}_M^{\text{Spatial}} = \sum_{i=0}^D \mathbf{M}_i \\ \mathbf{X} = \mathbf{A}_Q^{\text{Spatial}} \odot \mathbf{K} + \mathbf{A}_M^{\text{Spatial}} \odot \mathbf{K}. \end{cases} \quad (23)$$

Here, $\mathbf{A}_Q^{\text{Spatial}}$ represents a spatial attention vector of shape $1 \times N$ that models the importance of different spatial positions. $\mathbf{A}_M^{\text{Spatial}}$ denotes the importance of spatial positions computed from historical information, enabling the module to focus more on changing regions. The output \mathbf{X} of STA1 is obtained through the Hadamard product (\odot) between $\mathbf{A}_Q^{\text{Spatial}}$, $\mathbf{A}_M^{\text{Spatial}}$ and \mathbf{K} .

The computational process of STA2 follows a similar approach to STA1 but focuses on spectral–temporal feature fusion. The formulas are as follows:

$$\text{STA2} : \begin{cases} \mathbf{A}_Q^{\text{Spectral}} = \sum_{i=0}^N \mathbf{Q}_i \\ \mathbf{A}_M^{\text{Spectral}} = \sum_{i=0}^N \mathbf{M}_i \\ \mathbf{X} = \mathbf{A}_Q^{\text{Spectral}} \odot \mathbf{K} + \mathbf{A}_M^{\text{Spectral}} \odot \mathbf{K}. \end{cases} \quad (24)$$

Here, $\mathbf{A}_Q^{\text{Spectral}}$ denotes a spectral attention vector of shape $1 \times D$ that models the importance of different spectral channels, while $\mathbf{A}_M^{\text{Spectral}}$ represents the importance of spectral channels computed from historical information. The output \mathbf{X} of STA2 is obtained through the Hadamard product (\odot) among $\mathbf{A}_Q^{\text{Spectral}}$, $\mathbf{A}_M^{\text{Spectral}}$, and \mathbf{K} .

In the memory-enhanced spiking Transformer block, we construct a dual-branch architecture, with one branch based on STA1 and the other based on STA2. By performing feature extraction through these two branches separately and subsequently fusing their output features through summation, we can extract comprehensive spatiotemporal–spectral features.

D. Spiking Difference Module

As shown in Fig. 2, the SDM takes multitemporal features $\mathbf{X} \in \mathbb{R}^{T \times N \times D}$ as input. This single input is used to generate two representations through a temporal shifting operation. One representation is the original feature sequence \mathbf{X} . The other is a temporally shifted version $\mathbf{X}' = \{\mathbf{X}_t, \mathbf{X}_1, \dots, \mathbf{X}_{t-1}\}$. The temporal differences $\Delta\mathbf{X} = \mathbf{X} - \mathbf{X}' = \{\Delta\mathbf{X}_{1,t}, \Delta\mathbf{X}_{2,1}, \dots, \Delta\mathbf{X}_{t,t-1}\}$ are then input into spiking neurons and linear layers to extract features. The formulas are as follows:

$$\Delta\mathbf{X}_{i,j} = \mathbf{X}_i - \mathbf{X}_j \quad (25)$$

$$\Delta\mathbf{X}' = \text{Linear}(\text{PPSN}(\Delta\mathbf{X})) \quad (26)$$

where \mathbf{X}_i represents the feature at time step i . SDM can enhance the representation of change features by performing difference operations on features across different timesteps, enabling SpikeHCD to better capture local temporal variations and thereby improving the accuracy of CD tasks.

E. Overall Loss

At the end of the SpikeHCD, the features from SDM are input to the classification layer to obtain the probabilities of change and no-change occurrences. The CD task can be regarded as a binary classification problem; we employ cross-entropy loss [41] to optimize the network. The formulas are as follows:

$$p = \text{Linear}(\Delta\mathbf{X}') \quad (27)$$

$$L_{\text{cls}} = -\frac{1}{n} \sum_{i=1}^n (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)). \quad (28)$$

Here, n denotes the total number of samples, L_{cls} represents the classification loss, where y_i denotes the binary class label, and p indicates the probability output from the linear layer $\text{Linear}(\cdot)$. We train the network by backpropagating the gradients of loss L_{cls} through both spatial and temporal domains (BPTT) [42].

IV. EXPERIMENTAL EVALUATION

A. Datasets

We evaluate SpikeHCD on three datasets: the Farmland, River, and Hermiston datasets. The details of each dataset are as follows.

Farmland: Farmland dataset [43] covers agricultural areas in Yancheng, Jiangsu, China. It was captured by the Hyperion sensor on Earth Observing-1 (EO-1) satellite on May 3, 2006 and April 23, 2007. The dataset contains 242 spectral bands covering a wavelength range of 0.4–2.5 μm , with a spatial resolution of 30 m and an image size of 450 \times 140 pixels. After removing noisy bands, 155 valid spectral bands were retained for experiments.

River: River dataset [44] covers a river region in Jiangsu Province, China. It was collected by the EO-1 sensor on May 3, 2013 and December 31, 2013. The image size is 463 \times 241 pixels. After removing noisy bands, 198 valid spectral bands were retained for experiments.

Hermiston: Hermiston dataset [45] was collected over irrigated farmland in Hermiston, OR, USA. It was captured by the Hyperion sensor on EO-1 in 2004 and 2007. The image size is 307 \times 241 pixels, with 154 spectral bands remaining after noise removal.

B. Evaluation Measures

To comprehensively evaluate the performance of SpikeHCD, we employ overall accuracy (OA), Kappa coefficient (KC), energy consumption (power), and parameter count (Param) for assessment. The definitions of these metrics are as follows:

$$\text{OA} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \quad (29)$$

TABLE I
PERFORMANCE COMPARISON WITH EXISTING METHODS ON THE FARMLAND DATASET

Methods	OA%	KC($\times 100$)	Parm (M)	Power (mj)
SVM [4]	93.26	84.12	-	-
CVA [3]	96.23	90.81	-	-
CDFFormer [16]	95.99	90.35	0.24	6.40
BTCDNet [22]	96.80	92.35	1.56	20.45
CSANet [17]	97.17	93.17	2.43	41.31
MFCEN [20]	97.25	93.34	76.84	74.06
SSTFormer [18]	97.32	93.50	1.26	22.17
GLAFormer [19]	97.54	94.06	6.52	82.43
AIWSEN [23]	97.95	95.04	12.21	17.24
MsFNet [21]	98.24	95.72	8.01	20.27
Ours	98.27	95.80	0.56	1.16

$$\text{Kappa} = \frac{\text{OA} - p_e}{1 - p_e} \quad (30)$$

$$p_e = \frac{(\text{TP} \times \text{FN}) + (\text{TP} \times \text{FP}) + (\text{TN} \times \text{FN}) + (\text{TN} \times \text{FP})}{N^2}. \quad (31)$$

Here, N denotes the total number of samples, true positive (TP) represents the count of pixels correctly classified as changed regions, true negative (TN) indicates the count of pixels correctly classified as unchanged regions, false positive (FP) represents the count of pixels erroneously classified as changed regions, and false negative (FN) refers to pixels misclassified as unchanged regions.

Energy consumption is a crucial evaluation metric in SNNs. It depends on the number of synaptic operations (SOP), and it can be calculated as follows:

$$\text{SOP}^l = R \times T \times \text{FLOP}_s^l. \quad (32)$$

Here, l represents a module in the network, R is the firing rate of the spiking neurons in this module, and T denotes the timestep length. FLOP_s^l indicates the floating-point operations of module l . For both artificial neural networks (ANNs) and SNNs, energy consumption is calculated as follows:

$$\text{Power}_{\text{ANN}} = E_{\text{MAC}} \times \sum_{i=1}^L \text{FLOP}^i \quad (33)$$

$$\text{Power}_{\text{SNN}} = E_{\text{AC}} \times \sum_{i=1}^L \text{SOP}^i \quad (34)$$

where $\text{Power}_{\text{ANN}}$ and $\text{Power}_{\text{SNN}}$ represent the energy consumption of ANN and SNN, respectively. L denotes the total number of layers in the network. Consistent with the prior work [26], [27], [29], we assume that the multiply-accumulate (MAC) and accumulate (AC) operations are implemented on 45-nm hardware [46], with $E_{\text{MAC}} = 4.6$ pJ and $E_{\text{AC}} = 0.9$ pJ.

C. Compared Methods and Experimental Setup

To validate the effectiveness of SpikeHCD, we conducted comparative experiments with multiple CD methods, including traditional machine learning methods (CVA [3] and SVM [4]) and recent state-of-the-art (SOTA) ANN-based methods (CSANet [17], CDFFormer [16], SSTFormer [18], GLAFormer

TABLE II
PERFORMANCE COMPARISON WITH EXISTING METHODS ON THE RIVER DATASET

Methods	OA%	KC($\times 100$)	Parm (M)	Power (mj)
CVA [3]	85.75	45.21	-	-
SVM [4]	93.18	52.67	-	-
CDFFormer [16]	95.60	68.44	0.24	3.38
SSTFormer [18]	96.03	71.65	1.27	22.85
CSANet [17]	96.03	72.93	2.45	41.46
GLAFormer [19]	96.89	79.66	6.53	82.90
MsFNet [21]	96.95	80.33	8.01	20.28
BTCDNet [22]	97.09	81.86	1.59	21.52
MFCEN [20]	97.31	82.22	76.92	47.27
AIWSEN [23]	97.41	83.48	12.48	24.84
Ours	97.61	84.09	2.05	3.67

TABLE III
PERFORMANCE COMPARISON WITH EXISTING METHODS ON THE HERMISTON DATASET

Methods	OA%	KC($\times 100$)	Parm (M)	Power (mj)
SVM [4]	88.31	63.87	-	-
CVA [3]	93.11	77.45	-	-
SSTFormer [18]	94.38	83.43	1.26	22.15
CSANet [17]	94.60	83.93	2.43	41.28
CDFFormer [16]	94.76	84.54	0.30	4.29
GLAFormer [19]	95.92	88.08	6.52	82.43
BTCDNet [22]	96.03	88.73	1.55	20.42
MsFNet [21]	97.07	91.38	8.01	20.27
AIWSEN [23]	97.18	91.79	12.21	17.08
MFCEN [20]	97.33	92.31	76.91	74.15
Ours	97.21	91.95	0.56	1.22

[19], MFCEN [20], BTCDNet [22], AIWSEN [23], and MsFNet [21]). We used the PyTorch platform [47] and NVIDIA GTX 3090 GPUs. Following the same settings as in other methods, we allocated 3% of each dataset for training, 2% for validation, and the remaining samples for testing. All compared methods were implemented using the same experiment configurations reported in their original papers. The SpikeHCD was trained using the AdamW optimizer with a learning rate of 0.0004, a batch size of 128, and 150 training epochs. The patch size was set to 3×3 pixels. The hyperparameter λ of MSA was initialized as a learnable parameter with a starting value of 0.2. The energy consumption Power in experiments represents the energy consumed for inferring 64 samples. For the experiment on the Farmland dataset (see Table I), the dimension of the embedded sequence is 128; for the River dataset (see Table II), it is 256; and for the Hermiston dataset (see Table III), it is 128. For the ablation experiments, the default dimension of the embedded sequence is 128.

D. Experimental Results

1) *Results Analysis for Farmland Dataset*: The visualization results and evaluation metrics of different methods on the Farmland dataset are shown in Fig. 3 and Table I, respectively. Among the traditional machine learning-based methods, both CVA and SVM exhibit critical errors in CD results, with many red pixels (FP) and some green pixels (FN) at the

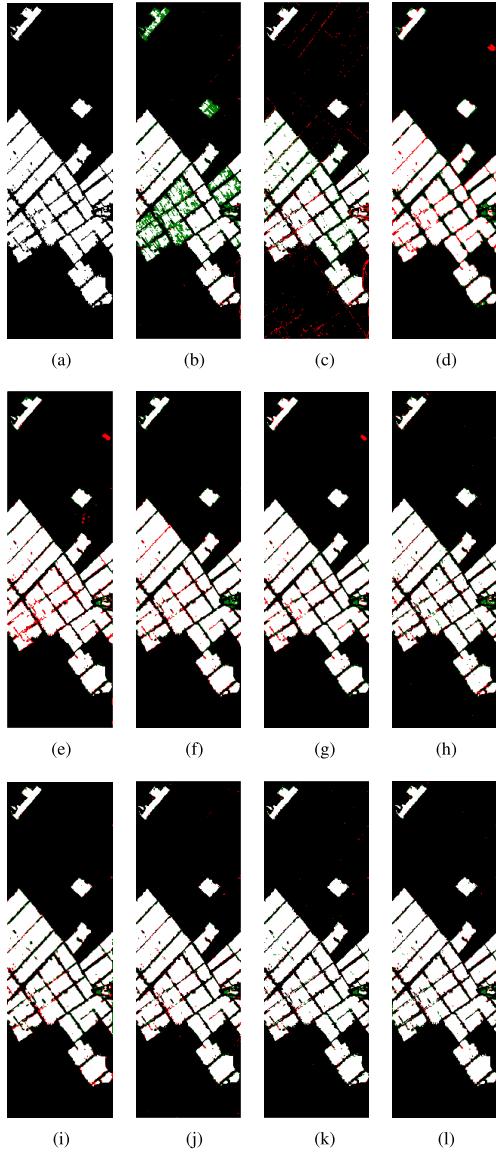


Fig. 3. Visualization results of different methods on the Farmland dataset, where TPs, TNs, FPs, and FNs are represented by white, black, red, and green colors, respectively. (a) Label. (b) SVM. (c) CVA. (d) CDFormer. (e) BTCDNet. (f) CSANet. (g) MFCEN. (h) SSTFormer. (i) GLAFormer. (j) AIWSEN. (k) MsFNet. (l) Ours.

edges of farmland areas. Specifically, SVM shows numerous green pixels in the center of the image. Consequently, SVM performs the worst result across all evaluation metrics, while CVA slightly outperforms SVM but still has many errors. The deep learning-based methods significantly surpass the traditional methods in both visualization and evaluation results. Compared with CDFormer, CSANet has fewer red pixels but more green pixels. BTCDNet achieves performance with 96.80% OA and 92.35 KC. Other methods, such as MFCEN and SSTFormer, demonstrate similar performance. GLAFormer achieves performance with 97.54% OA and 94.06 KC. AIWSEN achieves competitive results with 97.95% OA and 95.04 KC. MsFNet attains a high accuracy with 98.24% OA and 95.72 KC. Our proposed method achieves the highest accuracy (98.27% OA and 95.80 KC) while maintaining the lowest energy consumption (1.16 mJ) and minimal parameters (0.56 M).

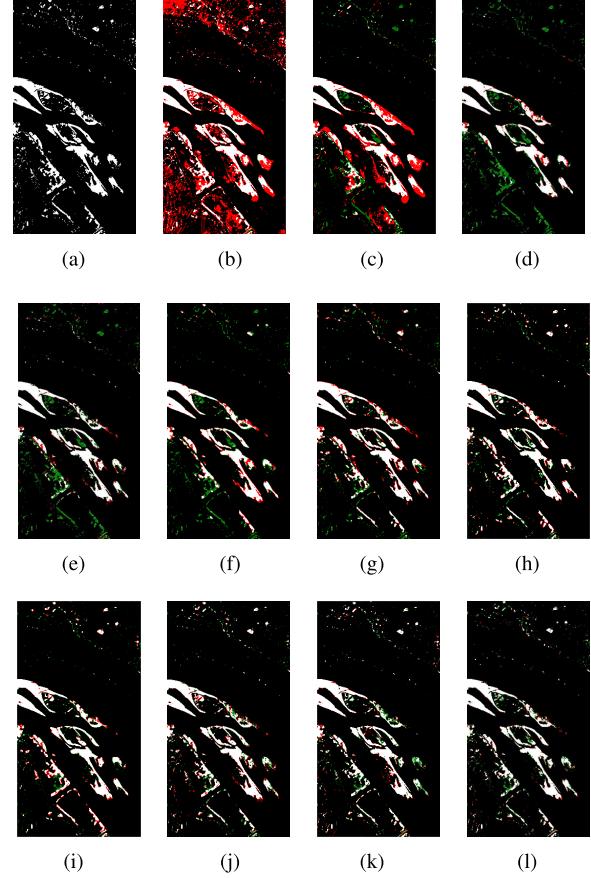


Fig. 4. Visualization results of different methods on the River dataset, where TPs, TNs, FPs, and FNs are represented by white, black, red, and green colors, respectively. (a) Label. (b) CVA. (c) SVM. (d) CDFormer. (e) SSTFormer. (f) CSANet. (g) GLAFormer. (h) MsFNet. (i) BTCDNet. (j) MFCEN. (k) AIWSEN. (l) Ours.

2) Results Analysis for River Dataset: The visualization results and evaluation metrics for different methods on the River dataset are shown in Fig. 4 and Table II, respectively. Among the traditional machine learning methods, the results of CVA and SVM contain many false pixels. The deep learning-based methods demonstrate relatively better results. Among them, CDFormer, SSTFormer, and CSANet show similar detection accuracy. GLAFormer performs better than these methods in OA and KC. Although MFCEN achieves a high OA (97.31%) and KC (82.22), its parameter count (76.92M) and energy consumption (47.27 mJ) are significantly higher than other methods. BTCDNet achieves 97.09% OA and 81.86 KC. MsFNet shows competitive performance (96.95% OA and 80.33 KC). AIWSEN demonstrates suboptimal KC performance (83.48) with 97.41% OA. Our proposed method achieves the highest OA (97.61%) and KC (84.09) among all methods, while maintaining efficient resource utilization with 2.05M parameters and the low energy consumption (3.67 mJ).

3) Results Analysis for Hermiston Dataset: The visualization results and evaluation metrics for different methods on the Hermiston dataset are shown in Fig. 5 and Table III, respectively. Among the traditional machine learning methods, both SVM and CVA exhibit critical errors in their CD results. The deep learning-based methods demonstrate

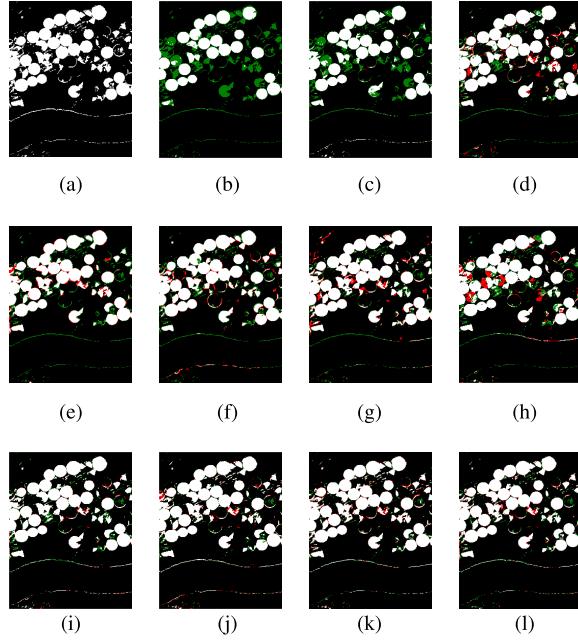


Fig. 5. Visualization results of different methods on the Hermiston dataset, where TPs, TNs, FPs, and FNs are represented by white, black, red, and green colors, respectively. (a) Label. (b) SVM. (c) CVA. (d) SSTFormer. (e) CSANet. (f) CDFormer. (g) GLAFormer. (h) BTCDNet. (i) MsFNet. (j) AIWSEN. (k) MFCEN. (l) Ours.

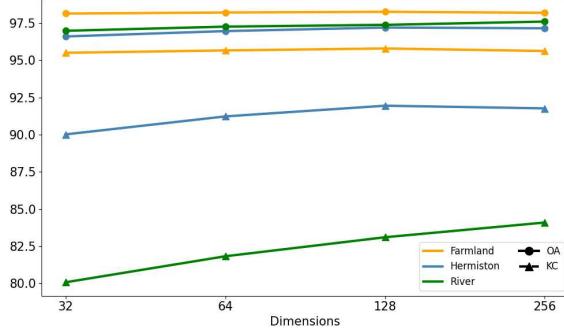


Fig. 6. Experimental results of SpikeHCD with different model sizes on the Farmland, Hermiston, and River datasets.

superior performance. SSTFormer, CSANet, and CDFormer show comparable detection accuracy. GLAFormer performs well but still has many green pixels (FN) along the river edges. BTCDNet shows competitive accuracy (96.03% OA and 88.73 KC). MsFNet achieves high performance (97.07% OA and 91.38 KC). AIWSEN shows better performance (97.18% OA and 91.79 KC). MFCEN achieves the highest OA (97.33%) and KC (92.31), but its parameter count (76.91M) and energy consumption (74.15 mJ) are substantially higher than other methods. The visualization results of our proposed method show fewer erroneous detections in the river boundary regions. Our method achieves the second-highest performance with 97.21% OA and 91.95 KC. Meanwhile, our method maintains a low parameter count (0.56M) and the lowest energy consumption (1.22 mJ).

4) Ablation Experiment:

a) *Ablation study on model size:* We evaluated the impact of model size (feature dimension ranging from 32

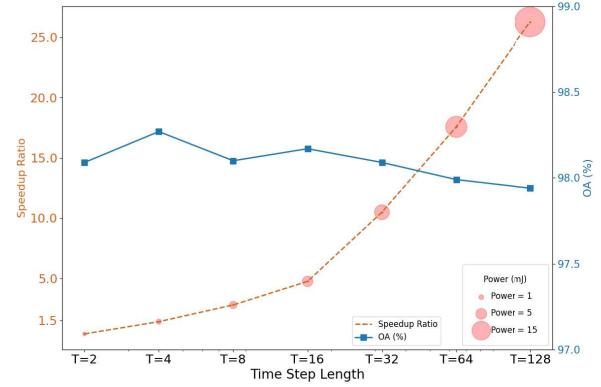


Fig. 7. Speedup ratio, OA, and energy consumption (power) on the Farmland dataset under different timestep lengths.

TABLE IV
PERFORMANCE COMPARISON WITH DIFFERENT SPIKING NEURONS
ON THE RIVER, FARMLAND, AND HERMISTON DATASETS

Methods	River		Farmland		Hermiston	
	OA%	Param (M)	OA%	Param (M)	OA%	Param (M)
LIF [38]	97.59	0.59	98.18	0.56	97.17	0.56
PSN [33]	97.12	0.59	98.00	0.56	96.57	0.56
PPSN (Ours)	97.39	0.59	98.27	0.56	97.21	0.56

to 256) on detection performance across the three datasets. As shown in Fig. 6, on the Farmland dataset, increasing the feature dimension from 32 to 128 improved both OA and KC; however, further expansion to 256 led to partial performance degradation, indicating overfitting. The results on the Hermiston dataset show similar trends, where performance peaked at 128 dimensions but declined slightly at 256 dimensions. On the River dataset, both OA and KC showed consistent improvements as the feature dimension increased from 32 to 256. At 32 dimensions, the model maintained lower parameter counts and energy consumption, albeit with marginally reduced accuracy compared to higher-dimensional variants. The configuration of 128 dimensions achieved an optimal balance between accuracy and energy efficiency, effectively capturing rich spatiotemporal-spectral features while avoiding computational redundancy from excessive dimensionality.

b) *Ablation study on PPSN:* To evaluate the performance of the PPSN, we replaced the spiking neurons in SpikeHCD with traditional LIF neurons and PSNs, and conducted comparisons across the three datasets. We also simulated conditions with timestep lengths ranging from 4 to 128 to comparatively analyze the inference time of PPSN and traditional LIF neurons. Fig. 7 and Table IV show the comparative experimental results of different methods.

As shown in Table IV, PPSN achieved higher OA than traditional LIF neurons on the Farmland and Hermiston datasets, while LIF slightly outperformed PPSN on the River dataset. First, the probability-driven reset mechanism in PPSN allows the membrane potential, which is originally directly reset to 0, to retain and transmit partial information to subsequent time steps. Second, the probability-driven reset enhances the robustness of the model. Therefore, PPSN demonstrates

TABLE V

ABLATION STUDY OF DUAL-BRANCH RESULTS ON THE RIVER, FARMLAND, AND HERMISTON DATASETS

Method	River		Farmland		Hermiston	
	OA%	KC(x100)	OA%	KC(x100)	OA%	KC(x100)
Spectral-temporal branch	97.24	82.28	98.20	95.65	96.88	90.92
Spatial-temporal branch	97.08	80.75	98.21	95.67	96.95	91.12
Dual branch (Ours)	97.39	83.10	98.27	95.80	97.21	91.95

superior performance compared with LIF. Compared with PSN, PPSN demonstrated better accuracy across all three datasets. By preserving the spatiotemporal dynamics of spiking neurons while enabling parallelization, PPSN exhibited better performance in CD tasks.

Fig. 7 illustrates the speedup ratio, OA and energy consumption (power) of SpikeHCD on the Farmland dataset under different timestep lengths. The speedup ratio is defined as the ratio of the inference time required by traditional LIF neurons to that required by our proposed PPSN, and is expressed as $(\text{time}_{\text{LIF}})/(\text{time}_{\text{PPSN}})$. A higher ratio indicates a greater improvement in computational efficiency by PPSN. As shown in Fig. 7, PPSN neurons outperform traditional LIF neurons. While traditional LIF neurons rely on iterative computation across timesteps, PPSN neurons leverage parallel matrix operations, which significantly enhance computational efficiency and enable more effective application in remote sensing tasks. When the timestep length $T = 4$, SpikeHCD achieves an OA of 98.27%, with an energy consumption of 1.16 mJ. As the timestep length increases, although SpikeHCD attains higher speedup ratio at larger time step lengths, both inference time and energy consumption increase accordingly. When the timestep length becomes too large, the gradient issue of the SNN tends to accumulate [48], thereby making it difficult to achieve better accuracy. We selected a time step length of $T = 4$ as it achieves competitive accuracy with lower energy consumption.

c) *Ablation study on dual-branch architecture:* To verify the effectiveness of the dual-branch structure, we designed ablation experiments to conduct a comparative analysis of the performance of three structures (spectral–temporal branch, spatial–temporal branch, and dual branch) across the three datasets.

The experimental results are shown in Table V. On the River dataset, the spectral–temporal branch achieved an OA of 97.24% and a KC of 82.28, respectively. In comparison, the spatial–temporal branch showed lower accuracy. The dual-branch structure achieved the best performance, increasing OA to 97.39% and KC to 83.10, respectively. On the Farmland dataset, the spectral–temporal and spatial–temporal branches performed similarly, with OA of 98.20%, 98.21%, and KC of 95.65, 95.67, respectively. The dual-branch structure achieved optimal results, reaching 98.27% OA and 95.80 KC, respectively. On the Hermiston dataset, the dual-branch structure has also demonstrated superior performance compared with other structures. By integrating spectral, spatial, and temporal information, the dual-branch structure fully leverages the multidimensional characteristics of hyperspectral data. It preserves

TABLE VI

ABLATION STUDY OF MSA RESULTS ON THE RIVER, FARMLAND, AND HERMISTON DATASETS

Methods	River			Farmland			Hermiston		
	OA (%)	KC (x100)	Param (M)	OA (%)	KC (x100)	Param (M)	OA (%)	KC (x100)	Param (M)
Spiking self-attention [29]	97.29	82.09	0.52	98.18	95.61	0.49	97.04	91.40	0.49
QK attention [32]	97.38	82.71	0.52	98.24	95.73	0.48	96.87	90.89	0.48
MSA (Ours)	97.39	83.10	0.59	98.27	95.80	0.56	97.21	91.95	0.56

TABLE VII

ABLATION STUDY OF PATCH SIZE RESULTS ON THE FARMLAND, HERMISTON, AND RIVER DATASETS

Patch Size	River		Farmland		Hermiston	
	OA%	KC(x100)	OA%	KC(x100)	OA%	KC(x100)
1 × 1	96.07	73.69	97.80	94.71	94.98	85.28
3 × 3	97.39	83.10	98.27	95.80	97.21	91.95
5 × 5	96.92	80.09	98.05	95.27	96.00	88.33
7 × 7	96.56	76.96	97.96	95.05	95.14	86.01
9 × 9	95.96	72.96	96.41	91.31	94.56	83.69

spatial details while enhancing the representation of spectral and temporal features, thereby improving CD accuracy.

d) *Ablation study on MSA:* In hyperspectral CD tasks, effective temporal modeling is critical for improving detection accuracy. To validate the effectiveness of the MSA module, we conduct ablation experiments comparing three attention mechanisms: Spiking self-attention, QK attention, and our proposed iterative attention MSA.

The experimental results are shown in Table VI. On the River dataset, the model using SSA achieved an OA of 97.29% and KC of 82.09, respectively. When replaced with Q-K-Attention, the performance improved to 97.38% OA and 82.71 KC, respectively. Our proposed MSA, maintaining similar parameter counts, demonstrated further improvements reaching 97.39% OA and 83.10 KC. On both the Farmland and Hermiston datasets, MSA consistently showed superior performance compared to other attention mechanisms. By efficiently incorporating historical information into attention interactions, MSA significantly enhances temporal modeling capacity, thereby improving the CD accuracy in hyperspectral data containing rich spatiotemporal information.

e) *Ablation study on patch size:* To evaluate the impact of input patch size on model performance, we conduct ablation experiments using five patch sizes (1×1 , 3×3 , 5×5 , 7×7 , and 9×9) on the Farmland, Hermiston, and River datasets.

The experimental results are shown in Table VII. On the Farmland dataset, the 3×3 patch size achieved the highest OA of 98.27%, while larger patch sizes showed progressively lower performance. Similarly, on the Hermiston dataset, accuracy decreased from 97.21% at 3×3 to 94.56% at 9×9 . On the River dataset, the 3×3 patch size achieved optimal accuracy of 97.39%, with performance declining as patch size increased. The 1×1 patch size consistently achieved lower performance than 3×3 patches across the three datasets, as it relies solely on spectral information while neglecting spatial features. Larger patch size introduced redundant spatial information, leading to suboptimal performance. The 3×3 patch size achieved better results by effectively balancing local spatial feature with spectral information.

TABLE VIII

ABLATION STUDY OF TEMPORAL DIFFERENCE STRATEGIES IN SDM ON THE RIVER, FARMLAND, AND HERMISTON DATASETS

Strategy	Farmland		Hermiston		River	
	OA%	KC(x100)	OA%	KC(x100)	OA%	KC(x100)
(1)	97.61	94.63	96.65	89.74	96.39	80.92
(2)	97.75	94.55	96.61	89.68	96.78	78.42
(3)	98.12	95.24	96.87	90.15	96.97	81.87
(4)	97.70	94.52	96.34	89.29	96.31	80.65
Baseline	98.27	95.80	97.21	91.95	97.39	83.10

f) Ablation study on SDM module: To evaluate the impact of temporal difference computational strategies in the SDM, we conduct ablation experiments on the Farmland, Hermiston, and River datasets. The baseline SDM employs a hybrid difference strategy combining long-range and short-range temporal differences: $\{\Delta X_{1,4}, \Delta X_{2,1}, \Delta X_{3,2}, \Delta X_{4,3}\}$. The difference $\Delta X_{i,j} = X_i - X_j$, where X_i represents the feature at time step i . We compare this with four different strategies: 1) $\{\Delta X_{1,3}, \Delta X_{2,4}\}$; 2) $\{\Delta X_{2,1}, \Delta X_{4,3}\}$; 3) $\{\Delta X_{1,4}, \Delta X_{2,3}\}$; and 4) $\{X_1, X_2, X_3, X_4\}$, which represents feature projection without temporal differencing. The results are shown in Table VIII.

As shown in Table VIII, experimental results demonstrate that our baseline difference strategy achieved superior performance compared to other strategies across the three datasets. Hyperspectral CD requires determining pixelwise changes, and it is crucial to consider diverse change features. Due to the removal of the temporal difference operation, method (4) achieved the worst performance. Our baseline SDM takes diverse difference features as input. The long-range $\Delta X_{1,4}$ difference represents globally significant changes between multitemporal HSIs; the short-range $\Delta X_{3,2}$ difference represents gradual changes between multitemporal HSIs; while the $\Delta X_{2,1}$ and $\Delta X_{4,3}$ differences represent critical spatial features learned by SNNs from two identical copies of HSIs. By incorporating all of these diverse features, SDM can extract richer and more robust features, thereby achieving superior CD performance.

V. CONCLUSION

This article proposes a spiking Transformer with parallel neurons and memory-enhanced attention named SpikeHCD, the first SNN specifically designed for hyperspectral CD, which effectively addresses the tradeoff between energy consumption and accuracy in existing methods. SpikeHCD optimizes the critical bottlenecks encountered when applying SNNs to hyperspectral CD tasks. First, we propose the PPSN to significantly enhance the computational efficiency, enabling more effective application of SpikeHCD in remote sensing tasks. Furthermore, we propose the MSA module and SDM to substantially improve the spatiotemporal-spectral modeling capacity and change feature extraction capability. Experimental results demonstrate that SpikeHCD not only exhibits significant advantages in low energy consumption and parameter count on multiple hyperspectral datasets but also achieves substantial efficiency improvements compared to traditional spiking neurons.

REFERENCES

- [1] F. Thonfeld, H. Feilhauer, M. Braun, and G. Menz, “Robust change vector analysis (RCVA) for multi-sensor very high resolution optical satellite data,” *Int. J. Appl. Earth Observ. Geoinf.*, vol. 50, pp. 131–140, Aug. 2016.
- [2] A. A. Nielsen, “The regularized iteratively reweighted MAD method for change detection in multi- and hyperspectral data,” *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 463–478, Feb. 2007.
- [3] W. A. Malila, “Change vector analysis: An approach for detecting forest changes with Landsat,” in *Proc. LARS Symposia*, 1980, p. 385.
- [4] F. Melgani and L. Bruzzone, “Classification of hyperspectral remote sensing images with support vector machines,” *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.
- [5] W. Dong, Y. Yang, J. Qu, S. Xiao, and Y. Li, “Local information-enhanced graph-transformer for hyperspectral image change detection with limited training samples,” *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5509814.
- [6] H. Sahbi, “Misalignment resilient CCA for interactive satellite image change detection,” in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 3326–3331.
- [7] W. Sun et al., “A label similarity probability filter for hyperspectral image postclassification,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 6897–6905, 2021.
- [8] B. Yang, Y. Mao, L. Liu, L. Fang, and X. Liu, “Change representation and extraction in stripes: Rethinking unsupervised hyperspectral image change detection with an untrained network,” *IEEE Trans. Image Process.*, vol. 33, pp. 5098–5113, 2024.
- [9] T. Zhan et al., “TDSSC: A three-directions spectral-spatial convolution neural network for hyperspectral image change detection,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 377–388, 2021.
- [10] X. Ou, L. Liu, B. Tu, G. Zhang, and Z. Xu, “A CNN framework with slow-fast band selection and feature fusion grouping for hyperspectral image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5524716.
- [11] L. Wang, L. Wang, Q. Wang, and L. Bruzzone, “RSCNet: A residual self-calibrated network for hyperspectral image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5529917.
- [12] R. Song, Y. Feng, C. Xing, Z. Mu, and X. Wang, “Hyperspectral image change detection based on active convolutional neural network and spatial-spectral affinity graph learning,” *Appl. Soft Comput.*, vol. 125, Aug. 2022, Art. no. 109130.
- [13] L. Wang, L. Wang, Q. Wang, and P. M. Atkinson, “SSA-SiamNet: Spectral-spatial-wise attention-based Siamese network for hyperspectral image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5510018.
- [14] Y. Yang, J. Qu, S. Xiao, W. Dong, Y. Li, and Q. Du, “A deep multiscale pyramid network enhanced with spatial-spectral residual attention for hyperspectral image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5525513.
- [15] C. Shi, Z. Zhang, W. Zhang, C. Zhang, and Q. Xu, “Learning multiscale temporal-spatial-spectral features via a multipath convolutional LSTM neural network for change detection with hyperspectral images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5529816.
- [16] J. Ding, X. Li, and L. Zhao, “CDFFormer: A hyperspectral image change detection method based on transformer encoders,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.
- [17] R. Song, W. Ni, W. Cheng, and X. Wang, “CSANet: Cross-temporal interaction symmetric attention network for hyperspectral image change detection,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.
- [18] Y. Wang et al., “Spectral-spatial-temporal transformers for hyperspectral image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5536814.
- [19] Z. Wang, F. Gao, J. Dong, and Q. Du, “Global and local attention-based transformer for hyperspectral image change detection,” *IEEE Geosci. Remote Sens. Lett.*, vol. 22, pp. 1–5, 2025.
- [20] J. Ding, X. Li, S. Xiang, and S. Chen, “Multilevel features fused and change information enhanced neural network for hyperspectral image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5502413.
- [21] Y. Feng, W. Ni, L. Song, and X. Wang, “MsFNet: Multi-scale fusion network based on dynamic spectral features for multi-temporal hyperspectral image change detection,” *Remote Sens.*, vol. 16, no. 16, p. 3037, Aug. 2024.
- [22] J. Luo, J. Li, X. Chu, S. Yang, L. Tao, and Q. Shi, “BTCDNet: Bayesian tile attention network for hyperspectral image change detection,” *IEEE Geosci. Remote Sens. Lett.*, vol. 22, pp. 1–5, 2025.

- [23] L. Wu, J. Peng, B. Yang, W. Sun, and Z. Ye, "AIWSEN: Adaptive information weighting and synchronized enhancement network for hyperspectral change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025, Art. no. 5503412.
- [24] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
- [25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2019, pp. 4171–4186.
- [26] Z. Zhou et al., "Spikformer: When spiking neural network meets transformer," in *Proc. 11th Int. Conf. Learn. Represent.*, 2022, pp. 2349–2365.
- [27] M. Yao et al., "Spike-driven transformer," in *Proc. Adv. Neural Inf. Process. Syst.*, 2023, pp. 64043–64058.
- [28] M. Yao et al., "Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips," in *Proc. 12th Int. Conf. Learn. Represent.*, 2024, pp. 1781–1803.
- [29] C. Zhou et al., "Spikingformer: Spike-driven residual learning for transformer-based spiking neural network," 2023, *arXiv:2304.11954*.
- [30] C. Zhou et al., "Enhancing the performance of transformer-based spiking neural networks by SNN-optimized downsampling with precise gradient backpropagation," 2023, *arXiv:2305.05954*.
- [31] X. Shi, Z. Hao, and Z. Yu, "SpikingResformer: Bridging ResNet and vision transformer in spiking neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 5610–5619.
- [32] X. Zhou et al., "QKFormer: Hierarchical spiking transformer using Q-K attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 37, 2024, pp. 13074–13098.
- [33] W. Fang et al., "Parallel spiking neurons with high efficiency and ability to learn long-term dependencies," in *Proc. Adv. Neural Inf. Process. Syst.*, 2023, pp. 53674–53687.
- [34] Y. Wang, K. Shi, C. Lu, Y. Liu, M. Zhang, and H. Qu, "Spatial-temporal self-attention for asynchronous spiking neural networks," in *Proc. 32nd Int. Joint Conf. Artif. Intell.*, Aug. 2023, pp. 3085–3093.
- [35] M. Yao et al., "Temporal-wise attention spiking neural networks for event streams classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10221–10230.
- [36] R.-J. Zhu, M. Zhang, Q. Zhao, H. Deng, Y. Duan, and L.-J. Deng, "TCJA-SNN: Temporal-channel joint attention for spiking neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 3, pp. 5112–5125, Mar. 2025.
- [37] M. Yao et al., "Attention spiking neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 9393–9410, Aug. 2023.
- [38] C. Teeter et al., "Generalized leaky integrate-and-fire models classify multiple neuron types," *Nature Commun.*, vol. 9, no. 1, p. 709, Feb. 2018.
- [39] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, no. 4, p. 500, 1952.
- [40] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1569–1572, Nov. 2003.
- [41] A. Mao, M. Mohri, and Y. Zhong, "Cross-entropy loss functions: Theoretical analysis and applications," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 23803–23828.
- [42] Y. Wu, L. Deng, G. Li, J. Zhu, and L. Shi, "Spatio-temporal backpropagation for training high-performance spiking neural networks," *Frontiers Neurosci.*, vol. 12, May 2018, Art. no. 323875.
- [43] Y. Yuan, H. Lv, and X. Lu, "Semi-supervised change detection method for multi-temporal hyperspectral images," *Neurocomputing*, vol. 148, pp. 363–375, Jan. 2015.
- [44] Q. Wang, Z. Yuan, Q. Du, and X. Li, "GETNET: A general end-to-end 2-D CNN framework for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 3–13, Jan. 2019.
- [45] M. Hasanlou and S. T. Seydi, "Hyperspectral change detection: An experimental comparative study," *Int. J. Remote Sens.*, vol. 39, no. 20, pp. 7029–7083, Oct. 2018.
- [46] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 10–14.
- [47] A. Paszke et al., "Automatic differentiation in PyTorch," in *Proc. NIPS Workshop on Autodiff*, 2017.
- [48] Y. Huang et al., "CLIF: Complementary leaky integrate-and-fire neuron for spiking neural networks," in *Proc. 41st Int. Conf. Mach. Learn.*, Jul. 2024, pp. 19949–19972.



Zihao Mei is currently a Researcher of computer science with Ningbo University, Ningbo, China. His research interests include spiking neural networks, neuromorphic computing, and hyperspectral image processing.



Jianhao Li received the master's degree from Ningbo University, Ningbo, China, in 2020, where he is currently pursuing the Ph.D. degree.

His research interests include spiking neural network, image processing, artificial intelligence, computer vision, hashing, and data mining.



Bolin Zhang received the Ph.D. degree in computer science from Hunan University, Changsha, China, in 2024.

He was funded by China Scholarship Council to participate in a joint Ph.D. Program at Nagoya University, Nagoya, Japan, from 2022 to 2023. Since July 2024, he has been with the Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, China, where he is currently a Lecturer. His main research interests are in neuromorphic computing, hashing retrieval, multimodal retrieval, and large multimodal models.



Chong Wang (Member, IEEE) received the Ph.D. degree from The University of Hong Kong, Hong Kong, in 2014.

He was a Research Assistant with The University of Hong Kong, and was a Research Associate with The Hong Kong Polytechnic University, Hong Kong. Since 2015, he has been with the Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, China, where he is currently an Associate Professor. His main research interests are in neuromorphic computing, video anomaly detection, neural network compression, human-computer interaction, and few-shot and zero-shot learning.



Lijun Guo (Member, IEEE) received the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2011.

He is currently a Full Professor with Ningbo University, Ningbo, China. His research focuses on intelligent computing, computer vision, neuromorphic computing, and medical image analysis, particularly in areas such as pedestrian reidentification, pose estimation, sparse representation with graph models, and intelligent image analysis for diagnostic support.



Jiangbo Qian (Member, IEEE) received the Ph.D. degree in computer science from Southeast University, Nanjing, China, in 2006.

He was a Visiting Scholar with the Department of Computer and Information Science, The University of Michigan, Ann Arbor, MI, USA. He is currently a Professor and a Vice Dean with the Graduate School, Ningbo University, Ningbo, China. He has published more than 200 papers in reputable international journals and conferences, including IEEE TRANSACTIONS. His research interests include image processing, computer vision, neuromorphic computing, database management, deep learning, hardware/software co-design, and hyperspectral image processing.