NEURAL DYNAMICS SELF-ATTENTION FOR SPIKING TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Integrating Spiking Neural Networks (SNNs) with Transformer architectures offers a promising pathway to balance energy efficiency and performance, particularly for edge vision applications. However, existing Spiking Transformers face two critical challenges: i) a substantial performance gap relative to their Artificial Neural Network (ANN) counterparts, and ii) considerable memory overhead. Our theoretical analysis and empirical evidence indicate that these limitations arise from the unfocused global attention paradigm of Spiking Self Attention (SSA) and the storage cost of large attention matrices. Inspired by the localized receptive fields and membrane potential dynamics of biological visual neurons, we propose LRF-Dyn, which enables attention computation via spiking neurons endowed with localized receptive fields. Specifically, we integrate a LRF mechanism into SSA, enabling the model to allocate greater attention to neighboring regions and thereby enhance local modeling capacity. Moreover, LRF-Dyn approximates the charge–fire–reset dynamics of spiking neurons within the LRF-SSA, substantially reducing memory requirements during inference. Extensive experiments on visual tasks confirm that our method lowers memory overhead while delivering significant performance improvements. These results establish LRF-Dyn as a key component for achieving energy-efficient Spiking Transformers.

1 INTRODUCTION

Vision Transformers (ViTs) (Vaswani et al., 2017; Liu et al., 2021; Yu et al., 2022) have achieved remarkable breakthroughs in computer vision tasks, including image classification (Chen et al., 2021; Deng et al., 2009), object detection (Carion et al., 2020; Li et al., 2022), and semantic segmentation (Zhou et al., 2017; Xie et al., 2021; Yu et al., 2018). As the core of ViTs, the self-attention mechanism computes query–key similarity through dot-product operation followed by softmax operation, incurring quadratic computational and memory costs with respect to the sequence length N (Yang et al., 2023). To address this issue, recent methods such as Linear attention (Katharopoulos et al., 2020; Zhang et al., 2024b) approximate or eliminate the softmax operation to explicitly compute the entire N^2 attention matrix, thus reducing memory usage and achieving linear-time complexity. However, these methods still rely on full-precision matrix multiplications, which incur substantial energy overhead and ultimately hinder their deployment on resource-constrained devices.

Spiking Neural Networks (SNNs) (Maass, 1997; Gerstner & Kistler, 2002; Masquelier et al., 2008) have attracted increasing attention due to their biological plausibility and potential for low-power computing. As the fundamental computational units, spiking neurons fire spikes only upon activation and remain silent otherwise, thereby enabling event-driven computation (Deng et al., 2020). This mechanism ensures sparse information transmission and effectively avoids redundant multiply-accumulate (MAC) operations, leading to reduced energy consumption and lower computational overhead (Caviglia et al., 2014; Roy et al., 2019). Moreover, several studies (Fang et al., 2021a; Hu et al., 2024) demonstrate that CNN-based SNNs can achieve performance comparable to that of their ANN counterparts while delivering substantial gains in energy efficiency. The combination of SNNs and Transformer architectures offers a potential pathway to balance energy efficiency and high performance (Zhou et al., 2023b; Huang et al., 2024b; Yao et al., 2025).

In recent years, several SNN-based Transformer architectures have been proposed, including Spikformer (Zhou et al., 2023b), QKFormer (Zhou et al., 2024), and Spike-Driven-v3 (Yao et al., 2025),

which significantly improve the performance of SNNs. Nevertheless, SNN-based Transformers still suffer from a performance gap and substantial memory overhead. As shown in Fig. 1(a), we compare the attention distributions of vanilla self-attention (VSA) and SSA: SSA exhibits discrete and global patterns, whereas VSA shows localized and sparse ones. This mismatch hinders SNNs from focusing on specific regions of information. Moreover, as illustrated in Fig. 1(b), SNNs can exploit matrix aggregation and flexibly adapt their computational paradigms to different application scenarios. Nevertheless, they still necessitate the explicit storage of large attention matrices, such as $\mathbf{QK}$ matrices of size N^2 or $\mathbf{KV}$ attention matrices of size d^2, which severely constrains their deployment on resource-limited devices. Thus, balancing the low-energy advantage of SNNs with the demands for low memory overhead and high performance remains a critical open challenge.

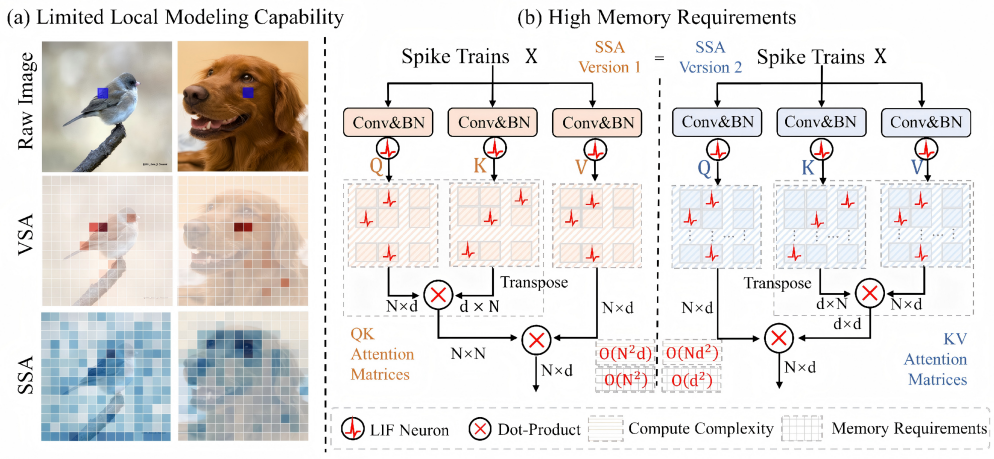

Figure 1: (a) Limited Local Modeling Capability: For a given n-th query (blue), VSA captures only limited and local relations, whereas SSA exhibits an almost global attention scope. (b) High Memory Requirements: SSA requires explicit storage of Queries ($\mathbf{Q}$), Keys ($\mathbf{K}$), Values ($\mathbf{V}$), and their associated attention scores ($\mathbf{QK}$ or KV), leading to substantial computational overhead.

Inspired by local receptive fields (Olshausen & Field, 1996; Gaynes et al., 2022) and membrane potential dynamics (Azouz & Gray, 2000) in biological vision, we propose LRF-Dyn, which enables attention computation via spiking neurons endowed with localized receptive fields. First, we theoretically and empirically compare VSA and SSA, showing that the lack of local modeling capacity in SSA accounts for its performance gap relative to VSA. To address this, we propose a Local Receptive Field SSA (LRF-SSA) method that introduces an LRF module into SSA, assigning higher weights to spatial neighbors to strengthen locality. Building on this, we introduce LRF-Dyn, which establishes an approximate correspondence between normalized self-attention aggregation and the charge–fire–reset process of spiking neurons, thus eliminating explicit attention-matrix storage. Extensive experiments across image classification and semantic segmentation tasks show that both LRF-SSA and LRF-Dyn significantly improve model performance, with LRF-Dyn further reducing memory consumption during inference. The main contributions are as follows:

- We identify two major challenges in the integration of SNNs and Transformer architectures, particularly in the context of SSA mechanisms: i) the performance gap caused by sparse and uniform attention distributions resulting from the removal of the softmax operation, and ii) the high memory overhead incurred due to the necessity of storing attention scores. These limitations hinder the balance between performance and computational resources.
- We propose LRF-Dyn to address these limitations. First, we introduce LRF-SSA by integrating LRF modules into SSA, assigning higher weights to spatially adjacent positions. Building on this, we propose LRF-Dyn, which establishes an approximate correspondence between neuronal membrane-potential dynamics and LRF-SSA, thereby eliminating the need for explicit attention-matrix storage and reducing memory overhead.
- Extensive experiments across diverse SNN architectures and visual tasks demonstrate that both LRF-SSA and LRF-Dyn enhance performance. Moreover, LRF-Dyn achieves lower memory requirements while preserving these performance gains, offering a practical solution for deployment in resource-constrained environments.

2 Related Work

Vision Transformer: ViT (Dosovitskiy et al., 2020; Liu et al., 2022b) converts images into patch-based tokens and utilizes self-attention to establish global contextual relationships, facilitating the selective aggregation of informative features (Naseer et al., 2021; Yang et al., 2021). However, these models suffer from $\mathcal{O}(\mathrm{N}^2)$ computational complexity and substantial memory overhead, which hinder scalability to large-scale visual tasks (Yang et al., 2023; Zhang et al., 2024a). To address these limitations, several studies (Choromanski et al., 2020; Guo et al., 2024) explore linear attention, which replaces the softmax operation with kernel-based approximations to reduce the complexity from $\mathcal{O}(\mathrm{N}^2)$ to $\mathcal{O}(\mathrm{N})$. These methods significantly reduce both computational and memory requirements, making them more suitable for high-resolution images (Shen et al., 2021; Guo et al., 2022b). However, existing models still rely on full-precision matrix multiplications, potentially increasing the model's energy consumption (Wang et al., 2020; Liu et al., 2022a).

Spiking Transformer: In recent years, researchers (Wang et al., 2023; Yao et al., 2023; Zhou et al., 2023a) have explored combining SNNs with Transformers to achieve a trade-off between energy efficiency and performance (Yao et al., 2024). Spikformer (Zhou et al., 2023b) introduces the SSA mechanism, which preserves spike-friendly properties while significantly improving the performance of SNNs. SpikingResformer (Shi et al., 2024) integrates ResNet with Transformer architectures to further reduce the model parameters. Furthermore, Spike-Driven-V3 (Yao et al., 2025) incorporates the Spike Frequency Approximation (SFA) mechanism into Spiking Transformers, enhancing their performance advantages. Although these models significantly reduce energy consumption, the inference process of SNN-based Transformers exhibits higher memory demands, limiting their deployment on resource-constrained devices (Aguirre et al., 2024).

3 Preliminary

3.1 Spiking Neuron Model

As the fundamental units of SNNs, spiking neurons (Izhikevich, 2003; Maass, 1997) receive presynaptic inputs and integrate them into the membrane potential, which is compared with the threshold to determine whether a spike is generated. Among them, the Leaky Integrate-and-Fire (LIF) neuron is the widely used model, whose dynamics are defined as follows:

$$\mathrm{U}[t+1] = \mathrm{H}[t] + \mathrm{W}S[t+1], \tag{1}$$

$$\mathrm{S}[t+1] = \Theta(\mathrm{U}[t+1] - \mathrm{V_{th}}), \tag{2}$$

$$\mathrm{H}[t+1] = \mathrm{V_{reset}}\mathrm{S}[t+1] + \tau \mathrm{U}[t+1](1 - \mathrm{S}[t+1]). \tag{3}$$

Here, $H[t]$ and $U[t]$ denote the pre-synaptic and post-synaptic membrane potentials, respectively, while $S[t]$ indicates the input spike at timestep t. W denotes the synaptic weight matrix. Spike generation is defined by the Heaviside function $\Theta(\cdot)$: if a spike occurs ($S[t+1] = 1$), $H[t]$ is reset to V_{reset}; otherwise, $U[t+1]$ decays with time constant τ and updates $H[t+1]$. For clarity, we denote the above process as $\mathrm{SN}\{\cdot\}$, which represents the dynamics of spiking neurons.

3.2 Spiking Self-attention Mechanisms

As the core component of the Spiking Transformer, the SSA mechanism captures spatio-temporal dependencies among tokens from spike trains and adaptively allocates importance across different regions. Given an input sequence $\mathbf{X} \in \mathbb{R}^{T \times B \times N \times D}$, it is projected through convolutional layers with distinct parameter matrices to obtain the Query $\mathbf{Q}$, Key $\mathbf{K}$, and Value $\mathbf{V}$ representations:

$$\mathbf{Q} = \mathrm{SN}\{\mathrm{BN}(\mathrm{Conv_Q}(\mathbf{X}))\}, \quad \mathbf{K} = \mathrm{SN}\{\mathrm{BN}(\mathrm{Conv_K}(\mathbf{X}))\}, \quad \mathbf{V} = \mathrm{SN}\{\mathrm{BN}(\mathrm{Conv_V}(\mathbf{X}))\}, \tag{4}$$

where $\mathrm{Conv}(\cdot)$ is the convolution operation, and $\mathrm{BN}(\cdot)$ represents batch normalization. Inspired by VSA mechanism, SSA computes the similarity via the dot product of $\mathbf{Q}$ and $\mathbf{K}$, and employs the resulting weights to aggregate $\mathbf{V}$. Specifically, the process of SSA is defined as follows:

$$\mathbf{Score} = \mathrm{s} \cdot \mathbf{Q} \times \mathbf{K}^{\top}, \quad \mathbf{Attn'} = \mathrm{Score} \times \mathbf{V}, \quad \mathbf{Attn} = \mathrm{SN}\{\mathrm{Attn'}\}, \tag{5}$$

s denotes the scaling factor. The attention output $\mathrm{Attn'}$ is processed by spiking neurons to ensure event-driven characteristics. Unlike VSA, SSA omits the softmax operation, thereby preserving the event-driven and spike-friendly characteristics of the attention mechanism.

4 Problem Analysis in Spiking Transformer

This section examines two main limitations of applying self-attention in SNNs: i) the omission of the softmax operation leads SSA to produce attention score distributions that deviate from those in VSA and ii) compared to other softmax-free attention variants, SSA introduces higher memory usage and inference overhead. Further details are provided in the following sections.

4.1 Limited Local Modeling Capability

As the core mechanism of ViT, the VSA mechanism computes attention scores through the dot-product operation followed by the softmax operation. Specifically, given Query $\mathbf{Q} \in \mathbb{R}^{N\times C}$ and $\mathbf{K} \in \mathbb{R}^{N\times C}$, the attention score $\mathbf{Attn_{vit}}$ can be defined as follows:

$$\mathbf{Attn_{vit}} = \mathbf{softmax}\left(\frac{\mathbf{Q}\times\mathbf{K}^T}{\sqrt{d}}\right), \quad \mathrm{attn_{vit}}[i] = \frac{\exp\{\mathrm{q_i k}_i\}}{\sum_{j=1}^{n}\exp\{\mathrm{q}_i\mathrm{k}_j\}}, \tag{6}$$

d denotes the features dimension, and $\mathrm{attn_{vit}}[i] \in \mathbb{R}^{1\times N}$ represents the attention score corresponding to between the i-th query $\mathrm{q_i}$ and remaining tokens. When q and k are more similar, the attention score is higher. Since neighboring tokens usually exhibit stronger similarity, ViT demonstrates superior local modeling ability (Yang et al., 2021). As shown in Fig. 2, 76.8% of the attention scores in ViT are concentrated at short Manhattan distances. In contrast, SSA produces an almost uniform distribution of attention scores. This mismatch constrains the local modeling capacity of SSA and hinders its ability to capture spatial similarities among neighboring regions.

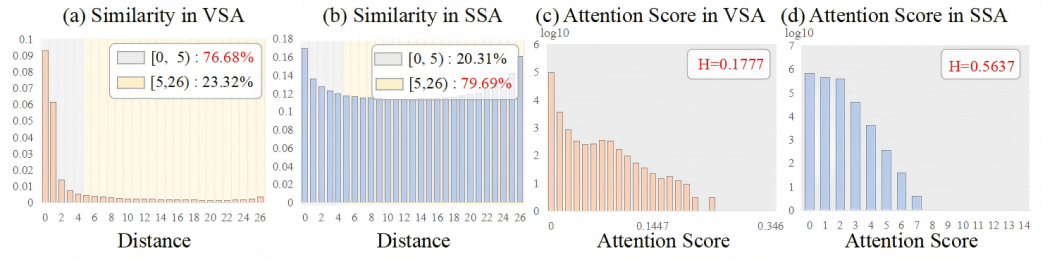

Figure 2: Mismatch between VSA and SSA attention scores: (a) and (b) show the average attention scores at different Manhattan distances, with VSA demonstrating stronger local modeling capabilities. (c) and (d) illustrate the distribution of attention scores, with VSA exhibiting lower entropy.

Furthermore, we compare the attention score distributions of SSA and VSA. As shown in Fig. 2, SSA exhibits a more uniform distribution, which hinders the model's ability to emphasize the relative importance of different regions. In contrast, VSA yields a lower-entropy distribution that focuses attention on a few critical regions, thereby enhancing feature extraction effectiveness.

4.2 High Memory Requirements during Inference

Similar to other softmax-free models (Wang et al., 2020), SSA leverages the associative property of matrix operations to reduce computational complexity to $\mathcal{O}(Nd^2)$. Nevertheless, this computational benefit is accompanied by a pronounced increase in memory overhead during inference. Specifically, for the query $\mathrm{q_n}[t]$ at timestep t, the attention score $\mathrm{attn_n}'[t]$ can be defined as follows:

$$\mathrm{attn}'_n[t] = \Big(\sum_{j=1}^{N}\mathrm{q}_n[t]\mathrm{k}_j[t]^T e_j^T\Big)\times \mathrm{v}[t] = \mathrm{q}_n[t]\times\sum_{j=1}^{N} k_j[t]^T v_j[t], \tag{7}$$

N denotes the total number of tokens, and e_j is a column vector of length N with the j-th entry set to 1 and all others set to 0. As shown in Eq. 7, for the n-th token, it is necessary to store not only the $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ matrices at each timestep, but also the intermediate results of the $\mathbf{KV}$ multiplication, leading to an additional $\mathcal{O}(\mathrm{d}^2)$ memory overhead. Notably, when $\mathrm{d} = 512$, SSA incurs substantial memory demands. These limitations substantially hinder its deployment on resource-constrained devices, particularly on neuromorphic chips (Davies et al., 2018; Pei et al., 2019).

5 METHOD

In this section, we propose the LRF-SSA method, which directly incorporates the LRF mechanism into SSA to enhance model performance. Furthermore, we restructure it from the perspective of neuron modeling, reducing memory overhead while preserving performance.

5.1 SPIKING SELF-ATTENTION WITH LOCAL RECEPTIVE FIELDS

To address the limitation of insufficient local receptive fields in the spiking self-attention paradigm, we introduce a local convolution module that enhances the mechanism's sensitivity to neighboring positions. Specifically, for the n-th token, its self-attention output $\mathrm{sattn}'_n[t]$ can be expressed as:

$$\mathrm{sattn}'_n[t] = \underbrace{\mathrm{q}_n[t] \times \sum_{j=1}^{N} k_j[t]^T v_j[t]}_{\text{Global Receptive Fields}} + \underbrace{\sum_{d} \sum_{i,j \in \Omega_d} r_{ij}^d \mathbf{V}^{\rho_k}}_{\text{Local Receptive Fields}}, \tag{8}$$

$\Omega_d = \{(i,j)|i,j \in \{-d,0,d\}\}$ represents the positional information of the neighboring region. To further reduce model complexity, we introduce multi-scale dilated convolutions to model local receptive fields. Specifically, two 3×3 depth-wise convolution kernels with dilation factors $d = 3$ and $d = 5$ are employed, where r_{ij} denotes the convolutional parameter at position (i,j).

Theorem 1 *Let $i \in \mathcal{N} = \{1, \cdots, n\}$ denotes the token position and defined the Manhattan distance between two elements as $\Delta = d(i,j) = |i-j|$. The normalized attention weight of VSA is $\alpha_{ij}^{vsa} \propto \exp(-\beta\Delta)$. For SSA, the weight satisfies $\alpha_{ij}^{ssa} \propto (\alpha - \beta\Delta)_+$. The LRF-SSA is defined as $\alpha_{ij}^{lrf\text{-}ssa} = (1-\lambda)\alpha_{ij}^{ssa} + \lambda r_{ij}$. Specifically, the expected receptive fields of LRF-SSA are defined:*

$$\mathbb{E}[\Delta_{\text{lrf-ssa}}] = (1-\lambda)\mu_{\text{ssa}} + \lambda\mu_r, \quad \text{where} \quad \mu_{\text{ssa}} = \mathbb{E}_{j \sim p_i^{\text{ssa}}}[\Delta_{\text{ssa}}], \quad \mu_r = \mathbb{E}_{j \sim p_i^{\text{r}}}(\Delta_r), \tag{9}$$

Since μ_{r} represents the receptive field around each token, it naturally satisfies that $\mu_{\text{r}} \leq \mu_{\text{ssa}}$. Theorem 1 demonstrates that LRF-SSA preserves the local attention characteristics similar to those of VSA, whereas SSA exhibits uniformly distributed attention weights. This difference arises because SSA removes the softmax operation, resulting in a linear decay of attention scores and diminishing its ability to distinguish between neighboring positions. In contrast, LRF-SSA incorporates additional learnable weights within the spatial domain, allowing the model to concentrate more effectively on information captured by the local receptive field. We need to further evaluate whether this operation can effectively preserve the low-entropy distribution property of the softmax operation.

Theorem 2 *For a given attention weights $x = (x_1, \ldots, x_N)$, the information entropy is defined as $H(x)$. In particular, the entropy of VSA is expressed as $H\big(\exp(-\beta\Delta)\big)$, while SSA satisfies $H\big((\alpha - \beta\Delta)_+\big)$. LRF-SSA satisfies $H\big((1-\lambda)p_{ij}^{ssa} + \lambda r_{ij}\big)$. These entropies satisfy the ordering:*

$$\mathrm{H}(p_i^{\text{lrf-ssa}}) \leq h(\alpha_i) + \alpha_i \mathrm{H}(p_i^{\text{ssa}}) + (1-\alpha_i)H(r_i) \leq \mathrm{H}(p_i^{\text{ssa}}) \quad \text{where} \quad 0 \leq \alpha_i \leq 1, \tag{10}$$

Theorem 2 demonstrates that LRF-SSA exhibits a lower-entropy distribution more closely aligned with VSA. This effect arises primarily from the presence of local receptive field modules, which amplify the differences among attention scores. A detailed proof is provided in the Appendix B and Appendix C. Compared with VSA, our method eliminates the need for the softmax operation, thereby achieving performance comparable to SSA while reducing computational cost.

In summary, our method effectively preserves the local receptive field and low-entropy distribution characteristics of VSA. Compared with SSA, it demonstrates stronger local receptive field capability while introducing only minimal additional overhead (two 3×3 convolution kernels and an extra $\mathcal{O}(\mathrm{d}^2)$ computational cost). The effectiveness will be further validated in the experimental section.

5.2 IMPLEMENTING SELF-ATTENTION THROUGH NEURONAL DYNAMICS

To further reduce the memory footprint and computational latency of SSA during inference, we restructure the LRF-SSA approach. As previously noted, the SSA module must store the Q, K, and V matrices at each timestep, along with their corresponding attention scores. Specifically, when the model applies the SSA Version 2 illustrated in Fig. 1(b), LRF-SSA reduces the computational complexity to $\mathcal{O}(\mathrm{Nd}^2)$, while requiring the additional storage of an attention matrix of size d^2.

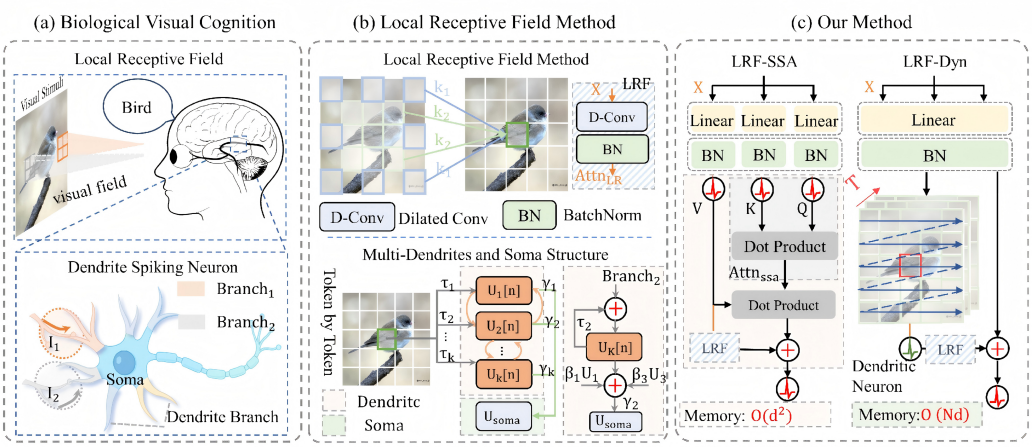

Figure 3: (a) Cognitive processes in biological vision, which exhibit local receptive field properties realized through multi-dendritic neurons. (b) The proposed LRF method together with the dynamic processes of dendritic neurons. (c) The implementation of LRF-SSA and LRF-Dyn.

Inspired by other softmax-free attention (Yang et al., 2023; Zhang et al., 2024b; Shen et al., 2021), LRF-SSA can be reformulated through causal inference to significantly reduce memory consumption. Specifically, Eq. 8 can be rewritten as follows:

$$\mathrm{sattn}_n[t]' = \mathrm{q}_n[t] \times \underbrace{\sum_{j=1}^{n-1} k_j[t]^T v_j[t]}_{\text{Memory Potential}} + \underbrace{k_n[t]^T v_n[t] + \sum_d \sum_{i,j \in \Omega_d} r_{ij}^d \mathrm{v}_{\rho_k}[t]}_{\text{Presynaptic Input}}, \tag{11}$$

In this manner, LRF-SSA method only needs to store $\sum_{j=1}^{n-1} k_j^\top v_j^\top$, thereby reducing the computational complexity to $\mathcal{O}(\mathrm{d}^2)$. The attention output is then multiplied by the current query vector $\mathrm{q_n}$ and transformed into a spike sequence through the SN layer. It closely parallels the charge–fire-reset dynamics of spiking neurons, where the first term represents membrane potential information and the second term represents presynaptic input. Additional implementation details of multi-dendritic neurons are described in Appendix D.

Therefore, we propose LRF-Dyn which leverages neuronal dynamics to formulate a novel paradigm for self-attention computation, whose dynamical process can be defined as follows:

$$\mathrm{X}_n[t] = \mathcal{A} \odot \mathrm{X}_{n-1}[t] + \mathbf{\Gamma} \mathrm{Token}^n[t], \quad \mathrm{sattn}'_n[t] = \mathrm{X}_n[t] + \sum_d \sum_{i,j \in \Omega_d} r_{ij}^d \cdot \mathrm{X}_{\rho_k}[t], \tag{12}$$

Here, $\mathrm{Token}_n[t]$ denotes the token input at position n. $\mathcal{A} \in \mathbb{R}^d$ denotes the decay factor, and $\mathbf{\Gamma} \in \mathbb{R}^d$ is defined as the membrane capacitance constant. Inspired by the multi-timescale behavior of photoreceptor neurons (Zheng et al., 2024; Chen et al., 2024), we define $\mathcal{A}$ and $\mathbf{\Gamma}$ in a dendritic form with local receptive fields to better allocate attention scores across tokens:

$$\mathcal{A} = \underbrace{\begin{bmatrix} c_1, \\ c_2, \\ \vdots \\ c_{n-1}, \\ c_n \end{bmatrix}^T}_{\mathcal{C}} \times \begin{bmatrix} -\frac{1}{\tau_1} & \beta_{2,1} & 0 & \cdots & 0 \\ \beta_{1,2} & -\frac{1}{\tau_2} & \beta_{3,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{1}{\tau_{n-1}} & \beta_{n,n-1} \\ 0 & 0 & \cdots & \beta_{n-1,n} & -\frac{1}{\tau_n} \end{bmatrix}, \quad \gamma = \begin{bmatrix} \gamma_1, \\ \gamma_2, \\ \vdots \\ \gamma_{n-1}, \\ \gamma_n \end{bmatrix}, \tag{13}$$

Here, d_n denotes the number of dendrites, and $\mathcal{C} \in \mathbb{R}$ represents the weights assigned to different dendrites. As shown in Fig. 3(b), for the n-th token, different dendritic branches produce distinct responses, which are further integrated by the soma through a specific mechanism to enhance spatial interactions and transform them into spike trains. Owing to the time-invariant property of the $\mathcal{A}$ matrix, the neuron can be trained efficiently following the (Chen et al., 2024). Compared with LRF-SSA, our model eliminates the SSA computation and only requires storing the membrane potential at each position, thereby substantially reducing memory usage during inference. We will further validate the effectiveness of this method in the experimental section.

5.3 Overall Architecture

By integrating local receptive fields with SSA, we propose the LRF-SSA method, which can be implemented through LRF-SSA and LRF-Dyn. Specifically, for an input spike train $x \in \mathbb{R}^{T\times N\times d}$, the computation of LRF-SSA is defined as follows:

$$\mathbf{Q},\mathbf{K},\mathbf{V} = \mathrm{SN}\{\mathrm{BN}(\mathrm{Conv}(\mathbf{X}))\}, \quad \mathbf{Score} = \mathrm{SN}\{\mathrm{s}\cdot(\mathbf{Q}\times\mathbf{K}^T + \sum\nolimits_{i,j\in\Omega_d} r_{ij}^d)\times\mathbf{V}\}. \tag{14}$$

Compared with SSA, LRF-SSA introduces almost no additional parameters, yet it significantly enhances the local modeling. Building on this, LRF-Dyn further reduces the memory requirements during model inference. The dynamics of LRF-Dyn are defined as follows:

$$\mathbf{H} = \mathcal{F}^{-1}\{\mathcal{F}(\mathbf{K})*\mathcal{F}(\mathbf{X})\}, \quad \mathbf{Score} = \mathrm{SN}\{\sum\sum\nolimits_{i,j\in\Omega_d} r_{ij}^d\cdot\alpha_k\mathbf{H}_{p_k(t)}\}, \tag{15}$$

where $\mathcal{F}^{-1}$ denotes the forward and inverse Fourier transforms, respectively, and $*$ represents the convolution operation. $\mathcal{K}(t)$ is the convolution kernel, defined as $\mathcal{C}\sum_{m=1}^{n-m}\mathcal{A}$. As shown in Fig. 3(c), both methods can be integrated into existing Transformer frameworks without any additional modifications. We further demonstrate the advantages of our approach in terms of both performance and memory efficiency in the experimental section.

6 Experiment

In this section, we compare with advanced SNN models on image classification (Deng et al., 2009) and semantic segmentation (Zhou et al., 2017) tasks. Additionally, we conduct ablation studies to validate the effectiveness of the proposed method, demonstrating its improved performance and reduced memory consumption. Detailed experimental results are provided below.

6.1 Image Classification & Semantic Segmentation

Image Classification Tasks: we evaluate both LRF-SSA and LRF-Dyn on the ImageNet-1k dataset. Specifically, we substitute the SSA mechanism in existing Spiking Transformers, including Spikformer (Zhou et al., 2023b), QKFormer (Zhou et al., 2024), and Spike-Driven-V3 (Yao et al., 2025), with the proposed LRF-SSA and LRF-Dyn. Furthermore, we compare our approach with recently advanced SNN models (Fang et al., 2021a; Hu et al., 2024; Yao et al., 2023; 2024).

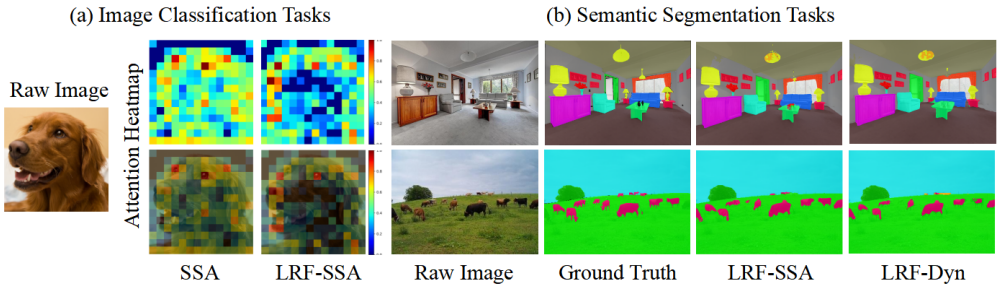

Figure 4: Visual results for image recognition and semantic segmentation. Both LRF-SSA and LRF-Dyn produce sparser attention scores and achieve finer-grained segmentation results.

As shown in Table 4, the proposed LRF-SSA method consistently delivers performance improvements across multiple architectures, with its two instantiations emphasizing different aspects. LRF-SSA primarily focuses on accuracy enhancement. On Spikformer, it improves accuracy by 1.24% and 0.85% at different parameter scales, while introducing fewer than 0.2M additional parameters. For QKFormer, LRF-SSA achieves an accuracy gain of 0.44% and 0.48% under the dim=384 and dim=512 configurations, respectively. Within the SDT-V3 architecture, LRF-SSA further demonstrates a favorable balance between efficiency and accuracy, attaining 76.22% recognition accuracy with only 5.24M parameters, substantially outperforming models of comparable size.

In contrast, LRF-Dyn preserves recognition performance while reducing storage complexity to $\mathcal{O}(\mathrm{kd})$, where k denotes the number of dendrites. For instance, on SDT-V3, LRF-Dyn achieves

a gain of 0.82% and 0.44% while requiring significantly less inference memory. Moreover, compared with CNN-based SNN models (Fang et al., 2021a; Hu et al., 2024), LRF-Dyn further delivers substantial performance gains without relying on attention mechanisms. As illustrated in Fig. 4(a), both LRF-SSA and LRF-Dyn exhibit ViT-like attention patterns but with sparser distributions, enabling the models to focus more effectively on salient regions. These results further demonstrate the superior performance and reduced memory requirements of our LRF-SSA framework.

Table 1: Comparison with Similar Methods on ImageNet-1K.

Method	Architecture	SR.	Param.(M)	Acc.(%)
SEWResNet (Fang et al., 2021a)	SEW-ResNet-34	-	21.79	67.04
MSResNet (Hu et al., 2024)	MS-ResNet-34	-	21.80	69.42
SDT-V1 (Yao et al., 2023)	Spike-driven-8-512	$\mathcal{O}(Nd)$	29.68	74.57
SDT-V2 (Yao et al., 2024)	Meta-SpikeFormer-384	$\mathcal{O}(d^2)$	15.08	74.10
	Meta-SpikeFormer-512	$\mathcal{O}(d^2)$	55.35	79.70
Spikformer (Zhou et al., 2023b)	Spikformer-8-512	$\mathcal{O}(d^2)$	29.68	73.38
	Spikformer-8-768	$\mathcal{O}(d^2)$	66.34	74.81
Spikformer + LRF-SSA	Spikformer-8-512	$\mathcal{O}(d^2)$	29.71	**74.62** (↑**1.24**)
	Spikformer-8-768	$\mathcal{O}(d^2)$	66.53	**75.66** (↑**0.85**)
Spikformer + LRF-Dyn	Spikformer-8-512	$\mathcal{O}(kd)$	29.71	**74.51** (↑**1.13**)
	Spikformer-8-768	$\mathcal{O}(kd)$	66.53	**75.58** (↑**0.77**)
QKFormer (Zhou et al., 2024)	HST-10-384	$\mathcal{O}(d^2)$	16.47	78.80
	HST-10-512	$\mathcal{O}(d^2)$	29.08	82.04
QKFormer + LRF-SSA	HST-10-384	$\mathcal{O}(d^2)$	16.55	**79.24** (↑**0.44**)
	HST-10-512	$\mathcal{O}(d^2)$	29.18	**82.52** (↑**0.48**)
QKFormer + LRF-Dyn	HST-10-384	$\mathrm{O}(kd)$	16.44	**79.21** (↑**0.41**)
	HST-10-512	$\mathcal{O}(kd)$	29.18	**82.48** (↑**0.44**)
SDT-V3 (Yao et al., 2025)	Efficient-Transformer-S	$\mathcal{O}(d^2)$	5.11	75.30
	Efficient-Transformer-L	$\mathcal{O}(d^2)$	18.99	79.80
SDT-V3 + LRF-SSA	Efficient-Transformer-S	$\mathcal{O}(d^2)$	5.24	**76.22** (↑**0.92**)
	Efficient-Transformer-L	$\mathcal{O}(d^2)$	19.25	**80.31** (↑**0.51**)
SDT-V3 + LRF-Dyn	Efficient-Transformer-S	$\mathcal{O}(kd)$	5.24	**76.12** (↑**0.82**)
	Efficient-Transformer-L	$\mathcal{O}(kd)$	19.25	**80.24** (↑**0.44**)

* SR represents the storage complexity requirements during inference.

Semantic Segmentation: To evaluate the effectiveness of our methods, we further conducted experiments on more challenging segmentation tasks. In particular, we evaluate on the ADE20K dataset (Zhou et al., 2017), which consists of 20K training images and 2K validation images across 150 semantic categories. Following the experimental protocol of SDT-V3 (Yao et al., 2025), we evaluate our method on models with 5M and 19M parameters. As shown in Table 2, our approach achieves notable performance gains, improving by 2.6% and 2.2%, respectively. In addition, compared with ResNet (Yu et al., 2022) without attention modules, LRF-Dyn achieves higher performance (36.3% and 43.1%) while requiring fewer model parameters. As illustrated in Fig. 4, we further provide qualitative comparisons on selected examples. The proposed method produces more fine-grained segmentation results, whereas SSA tends to yield only localized segmentations. This observation provides additional evidence of the effectiveness of LRF-SSA.

Table 2: Performance of segmentation.

Model	**Para. (M)**	**Attn**	**T**	**MIoU(%)**
ResNet	15.5	✗	1	32.9
(Yu et al., 2022)	28.5	✗	1	36.7
PVT	28.2	✓	1	39.8
(Wang et al., 2021)	48.0	✓	1	41.6
SDT-V3	5.1 + 1.4	✓	4	33.6
(Yao et al., 2025)	18.99 + 1.4†	✓	4	41.3
SDT-V3	5.1 + 1.4	✓	4	**36.2** (↑**2.6**)
+ LRF-SSA	10.0 + 1.4	✓	4	**43.5** (↑**2.2**)
SDT-V3	5.24 + 1.4	✗	4	**36.3** (↑**2.7**)
+ LRF-Dyn	19.25 + 1.4	✗	4	**43.1** (↑**1.8**)

† Results reproduced by ourselves.

6.2 Enhanced Local Modeling Ability with Lower Memory Requirements

To further validate the effectiveness of our method, we follow (Guo et al., 2022a; Ding et al., 2022) to visualize the receptive fields. As shown in Fig. 5(a), LRF-SSA significantly enhances the local receptive field of SSA and exhibits ViT-like local modeling capability. Similarly, LRF-Dyn also demonstrates a strong local modeling capacity. These results show the effectiveness of our method.

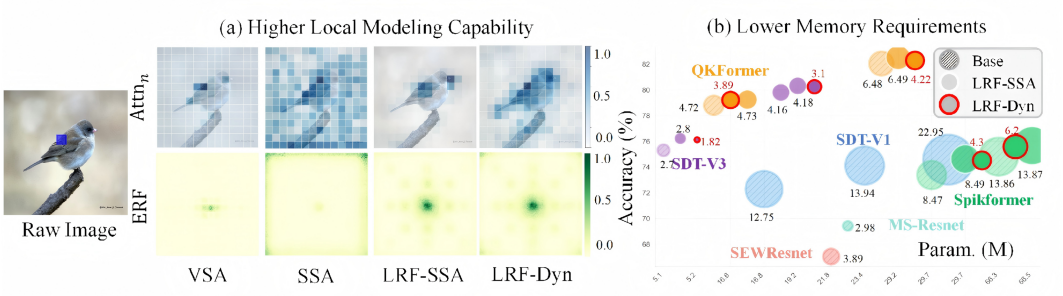

Figure 5: (a) Visualization of effective receptive field for different methods, where both LRF-SSA and LRF-Dyn demonstrate strong locality. (b) Comparative analysis of memory usage, accuracy, and parameter efficiency. The results show that LRF-Dyn maintains performance comparable to LRF-SSA while substantially reducing memory requirements during inference.

Furthermore, to compare the memory requirements of LRF-Dyn during inference, we visualize model accuracy and storage consumption at different scales. As illustrated in Fig. 5(b), Under the Spikformer-8-512 architecture, our method achieves a 1.13% increase in accuracy while simultaneously reducing memory usage by 49.4%. These results further confirm the effectiveness of our approach based on local receptive field enhancement and neuro-dynamics–inspired modeling.

6.3 Ablation Experiment

To further substantiate the effectiveness of the proposed approach, we conduct comprehensive experiments on the CIFAR-100 dataset. Specifically, we investigate the respective contributions of local capability modeling and neurodynamics-inspired self-attention to both model performance and memory consumption, using the Spikformer architecture as the evaluation framework.

For the LRF module, we systematically examine the impact of varying the number of convolution kernels on the results. Notably, without the LRF module, LRF-SSA is equivalent to the SSA method. As shown in Table 3, increasing the kernel count consistently improves recognition accuracy, demonstrating that the introduction of local receptive fields enhances the performance of both LRF-SSA and LRF-Dyn. Furthermore, by comparing the LRF-Dyn approach with a causal SSA model, we observe that LRF-Dyn consistently demonstrates improved performance under the same conditions. This further supports the effectiveness of our method.

Table 3: Ablation Experiment.

Method	w/o LRF	$\Omega \leq 1$	$\Omega \leq 3$	$\Omega \leq 5$
LRF-SSA	77.86	78.26	78.52	**78.64**
LRF-Dyn	77.78	78.16	78.50	78.57
Caused SSA†	74.30	75.30	76.20	76.50

† Results reproduced by ourselves.

7 Conclusion

In this paper, we analyze the challenges of integrating SSA into SNNs, focusing on two key limitations: the distinct attention distribution form compared to VSA, which limits further performance improvements, and the substantial memory overhead caused by the self-attention mechanism, which requires storing large attention weight matrices. To address these challenges, we propose LRF-Dyn, achieving an effective balance between performance and memory efficiency. First, we provide theoretical and empirical evidence for the importance of local modeling in self-attention, leading us to propose LRF-SSA, which incorporates a LRF module into SSA to enhance its local modeling capacity. Building on this, we approximate LRF-SSA with neuronal dynamics to remove the dependency on storing explicit attention matrices. This approach provides new theoretical insights and practical potential for deploying high-performance SNN models in resource-constrained edge environments.

REFERENCES

Fernando Aguirre, Abu Sebastian, Manuel Le Gallo, Wenhao Song, Tong Wang, J Joshua Yang, Wei Lu, Meng-Fan Chang, Daniele Ielmini, Yuchao Yang, et al. Hardware implementation of memristor-based artificial neural networks. *Nature communications*, 15(1):1974, 2024.

Rony Azouz and Charles M Gray. Dynamic spike threshold reveals a mechanism for synaptic coincidence detection in cortical neurons in vivo. *Proceedings of the National Academy of Sciences*, 97(14):8110–8115, 2000.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Stefano Caviglia, Maurizio Valle, and Chiara Bartolozzi. Asynchronous, event-driven readout of posfet devices for tactile sensing. In *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2648–2651. IEEE, 2014. doi: 10.1109/iscas.2014.6865717 .

Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 357–366, 2021.

Xinyi Chen, Jibin Wu, Chenxiang Ma, Yinsong Yan, Yujie Wu, and Kay Chen Tan. Pmsn: A parallel multi-compartment spiking neuron for multi-scale temporal processing. *arXiv preprint arXiv:2408.14917*, 2024.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, et al. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 38(1):82–99, 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Lei Deng, Yujie Wu, Xing Hu, Ling Liang, Yufei Ding, Guoqi Li, Guangshe Zhao, Peng Li, and Yuan Xie. Rethinking the performance comparison between snns and anns. *Neural networks*, 121:294–307, 2020.

Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11963–11975, 2022.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. *Advances in Neural Information Processing Systems*, 34:21056–21069, 2021a.

Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2661–2671, 2021b.

John A Gaynes, Samuel A Budoff, Michael J Grybko, Joshua B Hunt, and Alon Poleg-Polsky. Classical center-surround receptive fields facilitate novel object detection in retinal bipolar cells. *Nature Communications*, 13(1):5575, 2022.

Wulfram Gerstner and Werner M Kistler. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press, 2002.

Jialong Guo, Xinghao Chen, Yehui Tang, and Yunhe Wang. Slab: Efficient transformers with simplified linear attention and progressive re-parameterized batch normalization. *arXiv preprint arXiv:2405.11582*, 2024.

Jianyuan Guo, Kai Han, Han Wu, Yehui Tang, Xinghao Chen, Yunhe Wang, and Chang Xu. Cmt: Convolutional neural networks meet vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12175–12185, 2022a.

Meng-Hao Guo, Zheng-Ning Liu, Tai-Jiang Mu, and Shi-Min Hu. Beyond self-attention: External attention using two linear layers for visual tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5436–5447, 2022b.

Yifan Hu, Lei Deng, Yujie Wu, Man Yao, and Guoqi Li. Advancing spiking neural networks toward deep residual learning. *IEEE transactions on neural networks and learning systems*, 36(2):2353–2367, 2024.

Yulong Huang, Zunchang Liu, Changchun Feng, Xiaopeng Lin, Hongwei Ren, Haotian Fu, Yue Zhou, Hong Xing, and Bojun Cheng. Prf: Parallel resonate and fire neuron for long sequence learning in spiking neural networks. *arXiv preprint arXiv:2410.03530*, 2024a.

Zihan Huang, Xinyu Shi, Zecheng Hao, Tong Bu, Jianhao Ding, Zhaofei Yu, and Tiejun Huang. Towards high-performance spiking transformers from ann to snn conversion. In *Proceedings of the 32nd ACM international conference on multimedia*, pp. 10688–10697, 2024b.

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

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pp. 5156–5165. PMLR, 2020.

Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European conference on computer vision*, pp. 280–296. Springer, 2022.

Jing Liu, Zizheng Pan, Haoyu He, Jianfei Cai, and Bohan Zhuang. Ecoformer: Energy-saving attention with linear complexity. *Advances in Neural Information Processing Systems*, 35:10295–10308, 2022a.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12009–12019, 2022b.

Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29, 2016.

Wolfgang Maass. Networks of spiking neurons: the third generation of neural network models. *Neural networks*, 10(9):1659–1671, 1997.

Timothée Masquelier, Rudy Guyonneau, and Simon J Thorpe. Spike timing dependent plasticity finds the start of repeating patterns in continuous spike trains. *PloS one*, 3(1):e1377, 2008.

Muhammad Muzammal Naseer, Kanchana Ranasinghe, Salman H Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. *Advances in Neural Information Processing Systems*, 34:23296–23308, 2021.

Bruno A Olshausen and David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

Michalis Pagkalos, Spyridon Chavlis, and Panayiota Poirazi. Introducing the dendrify framework for incorporating dendrites to spiking neural networks. *Nature Communications*, 14(1):131, 2023.

Jing Pei, Lei Deng, Sen Song, Mingguo Zhao, Youhui Zhang, Shuang Wu, Guanrui Wang, Zhe Zou, Zhenzhi Wu, Wei He, et al. Towards artificial general intelligence with hybrid tianjic chip architecture. *Nature*, 572(7767):106–111, 2019.

Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784):607–617, 2019.

Shuaijie Shen, Chao Wang, Renzhuo Huang, Yan Zhong, Qinghai Guo, Zhichao Lu, Jianguo Zhang, and Luziwei Leng. Spikingssms: Learning long sequences with sparse and parallel spiking state space models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 20380–20388, 2025.

Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 3531–3539, 2021.

Xinyu Shi, Zecheng Hao, and Zhaofei Yu. Spikingresformer: Bridging resnet and vision transformer in spiking neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5610–5619, 2024.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Kexin Wang, Yuhong Chou, Di Shang, Shijie Mei, Jiahong Zhang, Yanbin Huang, Man Yao, Bo Xu, and Guoqi Li. Mmdend: Dendrite-inspired multi-branch multi-compartment parallel spiking neuron for sequence modeling. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27459–27470, 2025.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 568–578, 2021.

Ziqing Wang, Yuetong Fang, Jiahang Cao, Qiang Zhang, Zhongrui Wang, and Renjing Xu. Masked spiking transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1761–1771, 2023.

Yujie Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12:331, 2018.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal attention for long-range interactions in vision transformers. *Advances in Neural Information Processing Systems*, 34:30008–30022, 2021.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

Man Yao, Jiakui Hu, Zhaokun Zhou, Li Yuan, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer. *Advances in neural information processing systems*, 36:64043–64058, 2023.

Man Yao, JiaKui Hu, Tianxiang Hu, Yifan Xu, Zhaokun Zhou, Yonghong Tian, Bo XU, and Guoqi Li. Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=1SIBN5Xyw7.

Man Yao, Xuerui Qiu, Tianxiang Hu, Jiakui Hu, Yuhong Chou, Keyu Tian, Jianxing Liao, Luziwei Leng, Bo Xu, and Guoqi Li. Scaling spike-driven transformer with efficient spike firing approximation training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 325–341, 2018.

Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10819–10829, 2022.

Shuoxi Zhang, Hanpeng Liu, Stephen Lin, and Kun He. You only need less attention at each stage in vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6057–6066, 2024a.

Yu Zhang, Songlin Yang, Rui-Jie Zhu, Yue Zhang, Leyang Cui, Yiqiao Wang, Bolun Wang, Freda Shi, Bailin Wang, Wei Bi, et al. Gated slot attention for efficient linear-time sequence modeling. *Advances in Neural Information Processing Systems*, 37:116870–116898, 2024b.

Hanle Zheng, Zhong Zheng, Rui Hu, Bo Xiao, Yujie Wu, Fangwen Yu, Xue Liu, Guoqi Li, and Lei Deng. Temporal dendritic heterogeneity incorporated with spiking neural networks for learning multi-timescale dynamics. *Nature Communications*, 15(1):277, 2024.

Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6881–6890, 2021.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019.

Chenlin Zhou, Liutao Yu, Zhaokun Zhou, Zhengyu Ma, Han Zhang, Huihui Zhou, and Yonghong Tian. Spikingformer: Spike-driven residual learning for transformer-based spiking neural network. *arXiv preprint arXiv:2304.11954*, 2023a.

Chenlin Zhou, Han Zhang, Zhaokun Zhou, Liutao Yu, Liwei Huang, Xiaopeng Fan, Li Yuan, Zhengyu Ma, Huihui Zhou, and Yonghong Tian. Qkformer: Hierarchical spiking transformer using qk attention. *arXiv preprint arXiv:2403.16552*, 2024.

Tianfei Zhou, Wenguan Wang, Ender Konukoglu, and Luc Van Gool. Rethinking semantic segmentation: A prototype view. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2582–2593, 2022.

Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, YAN Shuicheng, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer. In *The Eleventh International Conference on Learning Representations*, 2023b.

Maria Zontak and Michal Irani. Internal statistics of a single natural image. In *CVPR 2011*, pp. 977–984, 2011. doi: 10.1109/CVPR.2011.5995401.

A USE OF LARGE LANGUAGE MODEL

In the preparation of this manuscript, we employ a Large Language Model (LLM) solely to aid and polish the writing. The LLM is used exclusively for language polishing, grammar correction, and improving the clarity and readability of the text. All technical ideas, methods, experiments, and conclusions presented in this paper are original contributions of the authors.

B PROOF OF THEOREM 1

In this section, we analyze the problem from the perspective of expected distance. Under an identical distance-bias assumption, SSA exhibits an expected distance that is almost global when compared with VSA. The LRF-SSA method partially alleviates the impact of this global receptive field.

B.1 DEFINITION OF EXPECTED RECEPTIVE RADIUS

Given a sequence $x = \{x_1, x_2, \ldots, x_N\}$ of length N, we compute the expected receptive radius to measure the receptive field size under different attention paradigms, which is defined as follows:

$$\mu = \mathbb{E}[\Delta] = \sum_{j=1}^{N} \hat{\alpha}_{ij}\Delta, \qquad \Delta = d(i,j) = |i-j|, \tag{1}$$

where $\hat{\alpha}_{ij}$ denotes the normalized attention weight from position i to position j, and Δ represents the distance between i and j. Considering the spatial continuity of image sequences, we adopt the Manhattan distance as the metric. In this way, the expected receptive radius μ provides an effective measure of the receptive field across different attention paradigms. A larger value of μ indicates a larger effective receptive radius, corresponding to a more global receptive field.

B.2 COMPARISON OF VSA, SSA AND LRF-SSA

Empirical studies in natural image statistics have demonstrated that patch similarity decreases as spatial distance increases (Zontak & Irani, 2011). For simplicity, we model this property as a linear decay with respect to distance. Specifically, for a given pair (q_i, k_j), their similarity can be defined:

$$q_i^\top k_j \;\approx\; a - \beta\Delta, \tag{2}$$

Accordingly, for SSA, the expected receptive radius can be represented as follows:

$$\alpha_{ij}^{vsa} \;=\; \frac{q_i^\top k_j}{\sum_{j=1}^{N} q_i^\top k_j} \;\propto\; \exp\{-\beta|i-j|\}, \quad \mathbb{E}[\Delta] = \sum \Delta\alpha_{ij} = \Delta\frac{\exp\{-\beta\Delta\}}{\sum_{t=0}^{N}\exp\{-\beta t\}}, \tag{3}$$

As $n \to \infty$, the expected receptive radius of VSA can be defined as follows:

$$\mu_\infty^{vsa} = \frac{\exp(-\beta)}{1-\exp(-\beta)} = \Theta(1). \tag{4}$$

Therefore, VSA exhibits a localized receptive field. In contrast, for SSA, since the softmax operation is omitted, its attention weight α_{ij} and the corresponding expected receptive radius can be defined as:

$$\alpha_{ij}^{ssa} \;\propto\; (a-\beta\Delta)_+, \quad \mathbb{E}[\Delta] = \sum \Delta\alpha_{ij} = \frac{(\alpha-\beta\Delta)_+}{\sum_{t=0}^{N}(\alpha-\beta\Delta)_+} \tag{5}$$

As $n \to \infty$, the expected receptive radius of VSA can be defined as follows:

$$\mu_\infty^{ssa} = \frac{(N-1)(3\alpha-\beta(2N-1))}{3(2\alpha-\beta(N-1))} = \Theta(N). \tag{6}$$

Therefore, SSA exhibits a broader receptive radius than VSA.

As an SSA variant with local receptive fields, the attention weight of SSA is not only determined by α_{ij}, but also incorporates an additional contribution from r_{ij}. Hence, the LRF-SSA attention weight can be expressed as a convex combination:

$$\alpha_{ij}^{\text{lra-ssa}} = (1-\lambda)\,\alpha_{ij}^{\text{ssa}} + \lambda\, r_{ij}, \quad \mathbb{E}[\Delta_{\text{lra-ssa}}] = (1-\lambda)\,\mu_{\text{ssa}} + \lambda\,\mu_r, \tag{7}$$

where $r_{ij} > 0$ denotes the local weights within the neighborhood $\Omega_d = \{j : |i - j| \leq d\}$, and $\mu_r \leq \mu_{\text{ssa}}$. Therefore, LRF-SSA exhibits an effective receptive field no larger than that of SSA, and this receptive field becomes increasingly local as λ increases. In the extreme case where $\lambda > 1$, LRF-SSA tends toward a highly localized receptive field. Therefore, the receptive field radii of the three methods follow the relation:

$$\mu_{\text{VSA}} \leq \mu_{\text{LRF-SSA}} \leq \mu_{\text{SSA}} \tag{8}$$

Therefore, the proposed method exhibits local modeling capabilities more similar to SSA, while avoiding the use of the softmax operation.

C Proof of Theorem 2

In this appendix, we provide a detailed theoretical analysis of the entropy properties of different self-attention mechanisms. Benefiting from the softmax operation, VSA exhibits a sharp and low-entropy distribution, concentrating most of the attention mass on a few neighboring positions. In contrast, SSA produces a more dispersed distribution, leading to higher entropy.

C.1 Information entropy

From an information-theoretic perspective, the entropy of the attention distribution provides a natural measure of its uncertainty and sharpness. Specifically, for a given attention weight vector $x = (x_1, \ldots, x_n)$, the entropy is defined as:

$$H(x) = -\sum_{i=1}^{n} x_i \log x_i, \tag{9}$$

where we use natural logarithms. A larger entropy indicates a more uniform and smoother distribution of attention scores, whereas a smaller entropy corresponds to a more concentrated and sharper allocation of attention.

C.2 Entropy of VSA, SSA and LRF-SSA

As shown in Eq. 17, the similarity decays linearly with distance Δ. Therefore, for VSA, the induced truncated geometric distribution over distances (ignoring multiplicity) is:

$$P_{\text{vsa}}(\Delta) = \frac{\exp\{-\beta\Delta\}}{\sum_{t=0}^{N-1} \exp\{-\beta t\}} = \frac{(1-r)\, r^{\Delta}}{1 - r^N}, \qquad r = \exp\{-\beta\} \in (0, 1), \tag{10}$$

discretized over $\Delta = 0, \ldots, N-1$. The corresponding entropy admits the closed form:

$$H_{\text{vsa}}(N, r) = \log \frac{1 - r^N}{1 - r} - \frac{r}{1 - r} \cdot \frac{1 - N r^{N-1} + (N-1) r^N}{1 - r^N} \log r, \tag{11}$$

and in the infinite-length limit, we obtain:

$$H_{\text{vsa}}(\infty, r) = -\log(1 - r) - \frac{r}{1 - r} \log r = \mathcal{O}(1). \tag{12}$$

Thus VSA yields a bounded, low-entropy distribution independent of N.

In contrast to the truncated geometric distribution of VSA, SSA employs linearly decaying weights without softmax normalization. Specifically, the unnormalized score is modeled as:

$$w(\Delta) = \alpha - \beta\Delta, \qquad 0 \leq \beta \leq \tfrac{\alpha}{N-1}, \tag{13}$$

where the constraint ensures non-negativity across all positions. After normalization, the distance distribution becomes:

$$P_{\text{ssa}}(\Delta) = \frac{\alpha - \beta\Delta}{\alpha N - \beta \frac{N(N-1)}{2}}, \qquad \Delta = 0, \ldots, N-1, \tag{14}$$

which leads to the entropy:

$$H_{\text{ssa}}(N, \alpha, \beta) = -\sum_{\Delta=0}^{N-1} \frac{\alpha - \beta\Delta}{S_0} \left[\log(\alpha - \beta\Delta) - \log S_0\right], \qquad S_0 = \alpha N - \beta \tfrac{N(N-1)}{2}. \tag{15}$$

For the special case $\beta = 0$, P_{ssa} reduces to the uniform distribution and the entropy attains its maximum $H_{\text{ssa}} = \log N$. At the other extreme, when $\beta = \alpha/(N-1)$, the distribution becomes triangular and the entropy satisfies $H_{\text{ssa}} = \log N + \mathcal{O}(1)$. Therefore, unlike VSA, the entropy of SSA scales as:

$$H_{\text{ssa}} = \Theta(\log N), \tag{16}$$

indicating a high-entropy distribution that spreads broadly across the sequence and corresponds to a nearly global receptive field.

Finally, we analyze the entropy of LRF-SSA. LRF-SSA can be formulated as a convex combination of SSA and a more concentrated local distribution R_i:

$$P_i^{\text{lra-ssa}}(\Delta) = (1 - \lambda_i)\, P_i^{\text{ssa}}(\Delta) + \lambda_i R_i(\Delta), \qquad \lambda_i \in [0, 1]. \tag{17}$$

The entropy of LRF-SSA is thus:

$$H\big(P_i^{\text{lra-ssa}}\big) = -\sum_{\Delta=0}^{N-1} \Big[(1-\lambda_i)P_i^{\text{ssa}}(\Delta) + \lambda_i R_i(\Delta)\Big] \log \Big[(1-\lambda_i)P_i^{\text{ssa}}(\Delta) + \lambda_i R_i(\Delta)\Big]. \tag{18}$$

Although a closed-form solution is difficult to obtain, we can derive an information-theoretic upper bound by introducing an auxiliary Bernoulli variable $Z \sim \text{Bernoulli}(\lambda_i)$ and applying the chain rule of entropy:

$$H\big(P_i^{\text{lra-ssa}}\big) \le h(\lambda_i) + (1-\lambda_i)H(P_i^{\text{ssa}}) + \lambda_i H(R_i), \tag{19}$$

where $h(\cdot)$ denotes the binary entropy. Since R_i is designed to be more localized, we typically have $H(R_i) \le H(P_i^{\text{ssa}})$, which implies:

$$H\big(P_i^{\text{lra-ssa}}\big) \le H(P_i^{\text{ssa}}). \tag{20}$$

Therefore, LRF-SSA consistently produces lower entropy than SSA, and the degree of reduction is controlled by the mixing coefficient λ_i; in other words, LRF-SSA interpolates between the global high-entropy behavior of SSA and the localized low-entropy property of VSA, providing a flexible trade-off between global coverage and local sharpness.

D Multi-Dendritic Neuron Modeling

This section provides an overview of multi-dendritic neuron models, highlighting their dynamics and categorizing them into two main types based on how the soma integrates dendritic inputs. We then describe the details of the aggregated model used in this work, as well as the mathematical formalization of its dynamics. Finally, we present the training process and inference process.

D.1 Dynamics of Dendritic Neuron

Dendritic neurons (Zheng et al., 2024; Chen et al., 2024; Wang et al., 2025; Pagkalos et al., 2023) possess strong temporal computation abilities and nonlinear expressive power, offering clear advantages in sequential tasks. In contrast to conventional point neurons (e.g., LIF neurons (Izhikevich, 2003; Maass, 1997)), multi-dendritic neurons consist of two main components: the dendrites and the soma. The dendritic structure introduces heterogeneous temporal factors, allowing the extraction of temporal features across multiple scales. The dynamics of i-th dendrite can be defined as follows:

$$C_m \frac{dv_i}{dt} = -I_{Leak}(v_i) + g_{i-1,i}(v_{i-1} - v_i) + g_{i,i+1}(v_{i+1} - v_i) + I, \tag{21}$$

C_m denotes the membrane capacitance, I_{leak} represents the leak current of the membrane potential v_i, and $g_{i,j}(\cdot)$ characterizes the coupling relationship between compartment i and j. Subsequently, the soma integrates the outputs from the dendrites and converts them into spike signals.

D.2 Overview of Two Types of Dendritic Neuron

Based on the manner in which somatic input is processed, multi-dendritic neurons can be classified into two primary types. One type accumulates the output of only the last dendrite (Wang et al., 2025; Chen et al., 2024). This approach assumes that information transmission occurs only between adjacent dendrites. The other type integrates the outputs of all dendrites (Zheng et al., 2024), allowing the soma to store and process all temporal heterogeneity information. The somatic neural dynamics for these two types can be modeled as follows:

$$C_m \frac{dv_n}{dt} = -I_{Leak}(v_n) + g_{n-1,n}(v_{n-1} - v_n) + I, \tag{22}$$

$$C_m \frac{dv_n}{dt} = -\sum_{i}^{n} \gamma_i I_{Leak}(v_i) + I, \tag{23}$$

Eq.22 processes the output from only the n-th dendrite, assuming that information transmission occurs between adjacent dendrites. In contrast, Eq.23 integrates the outputs of all dendrites, ensuring that the soma can store and process all temporal heterogeneity information. In this work, we adopt the second type of neuron to achieve temporal modeling across multiple timescales.

D.3 Formalization of Multi-Dendritic Neuron Dynamics

To facilitate mathematical analysis, we formalize these two approaches as follows. Specifically, the neural dynamics of multi-dendritic neurons can be equivalently expressed as:

$$\frac{d\mathrm{V_d}}{dt} = \tau \mathrm{V_d}[t] + \Gamma I[t], \quad \frac{d\mathrm{V_s}}{dt} = -\frac{1}{\tau_s}(\mathrm{V}_s[t] - v_{rest}) + \mathcal{C}\mathrm{V}_d, \tag{24}$$

Here, $\Gamma \in \mathbb{R}^{1\times n}$ denotes the membrane capacitance constant, $v_d \in \mathbb{R}^{n\times n}$ represents the membrane potential accumulation process of dendritic components, and $v_s \in \mathbb{R}$ represents the accumulation process of the somatic membrane potential. When v_s exceeds the threshold, the dendritic neuron generates a spike and resets the somatic membrane potential. The parameters $\tau_d \in \mathbb{R}^{n\times n}$ and $\tau_s \in \mathbb{R}$ denote the decay factors of dendritic and somatic components, respectively.

Different dendritic neuron models correspond to distinct forms of τ_d and Γ. Specifically, τ_d determines the computational form of dendrites, while Γ represents the manner in which the soma integrates dendritic inputs. In this work, τ_d and Γ can be further formalized as:

$$\tau_d = \begin{bmatrix} -\frac{1}{\tau_1} & \beta_{2,1} & 0 & \cdots & 0 \\ \beta_{1,2} & -\frac{1}{\tau_2} & \beta_{3,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -\frac{1}{\tau_{n-1}} & \beta_{n,n-1} \\ 0 & 0 & \cdots & \beta_{n-1,n} & -\frac{1}{\tau_n} \end{bmatrix}, \quad \Gamma = \begin{bmatrix} \gamma_1, \\ \gamma_2, \\ \vdots \\ \gamma_{n-1}, \\ \gamma_n \end{bmatrix}, \tag{25}$$

n denotes the number of dendrites. To achieve efficient inference, we can discretize Eq. 25. Specifically, the neural dynamics of multi-dendritic neurons can be defined as follows:

$$\mathrm{V}_d[t] = \exp\{\delta\tau_d\}\mathrm{V}_d[t-1] + \Gamma I, \quad \mathrm{V}_s[t] = -\exp\{\delta\tau_s\}\mathrm{V}_d[t-1] + \mathcal{C}\mathrm{V}_t[t], \tag{26}$$

δ denotes the discrete time step. In this work, the output of the soma is fed into the SN layer to enable effective interaction along the temporal dimension.

D.4 Training Process of LRF-Dyn

Unlike typical sequential tasks such as LRA benchmark (Tay et al., 2020), in static image tasks, SNNs employ frequency coding (Wu et al., 2018; Fang et al., 2021b) to approximate the continuous activation values of ANNs. Specifically, the input is represented as $x \in \mathbb{R}^{T\times N\times C}$. In this work, our LRF-Dyn model primarily operates along the spatial dimension N. For clarity of exposition, we focus on the case $T = 1$, which provides a clearer explanation of the training process.

Given the high resolution of image-related tasks, where $n \geq 196$, the complexity of training dendritic neurons increases substantially. Consequently, achieving efficient training while preserving a

sequential inference paradigm remains a central challenge. To address this, some researchers (Chen et al., 2024; Shen et al., 2025; Wang et al., 2025; Huang et al., 2024a) employ Fourier transforms to markedly reduce training costs, thereby enabling models to handle long-sequence tasks. In particular, for the n-th $\mathrm{token}[n] \in \mathbb{R}^{1\times C}$, the cumulative membrane potential $\mathrm{V}_d[n]$ of its corresponding dendritic component can be defined as follows:

$$\mathrm{V}_d[n] = \sum_{i=0}^{n} \Gamma \exp\{i \cdot \delta\tau_d\} \cdot \mathrm{token}[n-i]. \tag{27}$$

Since τ_d is time-invariant, Eq. 27 can be reformulated as a convolution between the input signal X and the kernel K, thereby reducing the training complexity of the model from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$.

$$\mathrm{V}_d = (\mathrm{K} * \mathrm{Token}) = \mathcal{F}^{-1}\left\{\mathcal{F}\{\mathrm{K}\} \cdot \mathcal{F}\{\mathrm{Token}\}\right\}, \quad \mathrm{K} = \left[\exp\{1\cdot \delta\tau_d\},\ \ldots,\ \exp\{n\tau_d\}\right], \tag{28}$$

$\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ denote the forward and inverse Fourier transform operations. Subsequently, the soma integrates inputs from different dendrites and stores them in the form of a membrane potential. Specifically, the dynamics of dendritic neurons can be expressed as:

$$\mathrm{X}[n] = \mathcal{C}V_d[n], \quad \mathrm{V}_s[n] = \mathrm{X}[n] + \sum_{d} \sum_{i,j\in\Omega_d} r_{ij}^{d} \cdot \mathrm{X}[\rho_k], \quad \mathrm{S} = \mathrm{SN}(\mathrm{V_s}), \tag{29}$$

$\mathrm{V_s}$ defines the presynaptic input of the soma, which integrates both the information from the current patch and the interactions among neighboring tokens. It is passed to the SN layer, where it is converted into spike trains: when the membrane potential exceeds the threshold, the neuron emits a spike; otherwise, it is accumulated with the membrane potential from the previous timestep. This mechanism effectively enables interaction across both spatial and temporal dimensions.

D.5 Inference Process of LRF-Dyn

For the inference process, as discussed earlier, LRF-Dyn achieves lower memory requirements. Specifically, given $\mathrm{Token} \in \mathbb{R}^{T\times N\times d}$, the inference procedure of LRF-Dyn can be described:

$$\mathrm{X}_n[t] = \mathcal{A}\odot \mathrm{X}_{n-1}[t] + \Gamma \mathrm{Token}^n[t], \quad \mathrm{sattn}'_n[t] = \mathrm{X}_n[t] + \sum_{d}\sum_{i,j\in\Omega_d} r_{ij}^{d} \cdot \mathrm{X}_{\rho_k}[t], \tag{30}$$

$\mathcal{A} \in \mathbb{R}^d$ denotes the decay factor, and $\Gamma \in \mathbb{R}^d$ is defined as the membrane capacitance constant. It only requires storing $\mathrm{sattn}' \in \mathbb{R}^{T\times N\times d}$, thereby avoiding the need to store the attention matrix with high spatial complexity. As a result, the memory cost during computation is significantly reduced.

E Design of Hyperparameters

Table 4: Comparison of Hyperparameters for Different Model Architectures

Hyper-parameter	**Spikformer**	**QKformer**	**Spike-Driven V3**
Timestep	4	4	4
Epochs	100	200	200
Resolution	224	224	224
Batch size	64	100	600
Optimizer	AdamW	AdamW	LAMB
Base Learning rate	7e-6	6e-4	6e-4
Learning rate decay	Cosine	Layer-wise 1.0	Layer-wise 1.0
Warmup epochs	5	5	10
Weight decay	5e-2	5e-2	0.05
Rand Augment	rand-m9-mstd0.5-inc1	rand-m9-mstd0.5-inc1	rand-m9-mstd0.5-inc1
Mixup	0.8	0	0
Cutmix	1.0	0	0
Label smoothing	0.1	0.1	0.1

All experiments are conducted on the ImageNet-1K dataset using the PyTorch framework. Training is carried out on four NVIDIA A800 GPUs with distributed data parallelism, while inference-time

memory evaluations were performed on an NVIDIA GeForce RTX 4090 GPU with a batch size of 16. The detailed hyperparameter configurations for each architecture are provided in Table 4.

To ensure fair comparison across architectures, we adopt the open-source implementations of Spikformer, QKformer, and Spike-driven V3. In each case, only the self-attention modules are replaced, while all other components (e.g., patch embedding, MLP layers, normalization layers) remain unchanged. In addition, the number of dendrites is fixed at eight to maintain parameter consistency between LRF-SSA and LRF-Dyn. This controlled design allows us to isolate the impact of different attention mechanisms on model performance.

F Image Classification

To further demonstrate the effectiveness of our method, we visualized the heatmaps on the ImageNet dataset and compared our approach with the SSA method. We selected the first layer for visualization. In addition, since LRF-Dyn does not involve an explicit attention mechanism, it is not feasible to generate its attention heatmaps. Therefore, in this section, we mainly compare the attention results of LRF-SSA. The visualization results are shown in Fig. 6.

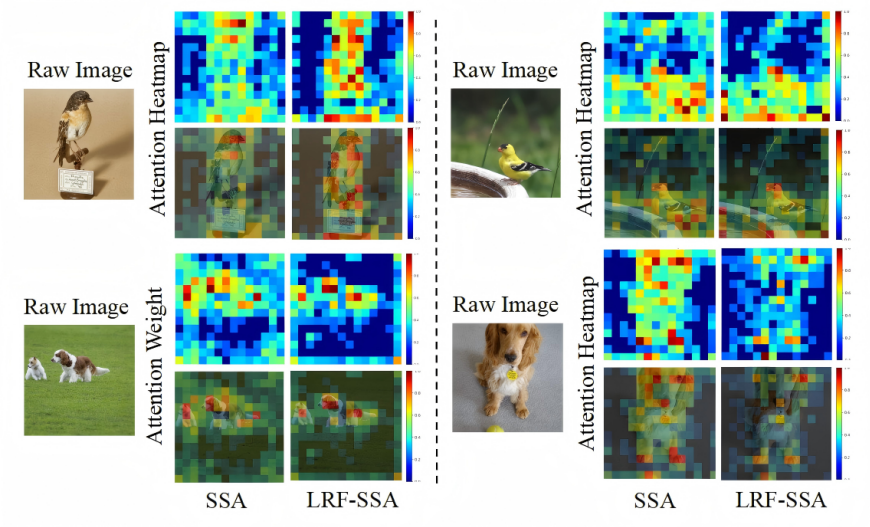

Figure 6: Attention heatmaps of LRF-SSA (Ours) with the SDT-V3 architecture on ImageNet. Our method demonstrates more consistent and sparse attention across diverse samples.

G Semantic Segmentation

In this study, we utilize the ADE20K semantic segmentation dataset (Zhou et al., 2017; 2019; 2022; Zheng et al., 2021), which contains more than 20,000 training images and 2,000 validation images with fine-grained pixel-level annotations for both objects and their components. The dataset is characterized by extensive semantic diversity, covering 150 categories that include environmental elements (e.g., sky, road, grass) as well as distinct objects (e.g., people, vehicles, furniture).

We adopt SDT-V3 (Yao et al., 2025), pretrained on ImageNet-1K, as the backbone and couple it with the Pyramid Vision Transformer (PVT)(Xie et al., 2021; Wang et al., 2021) for semantic segmentation. The additional parameters are initialized using the Xavier scheme. Training is conducted with a batch size of 20 over 160,000 iterations, employing the AdamW optimizer with an initial learning rate of 1×10^{-4} and a polynomial decay schedule (power 0.9). To ensure stable convergence, a linear warmup is applied during the first 150,000 iterations. Compared with the vanilla V3 model, our approach yields clear improvements on both 19m and 5m metrics. Furthermore, LRF-Dyn achieves substantially better performance than CNNs of comparable scale, even without relying on attention mechanisms. This result further underscores the effectiveness of our approach.

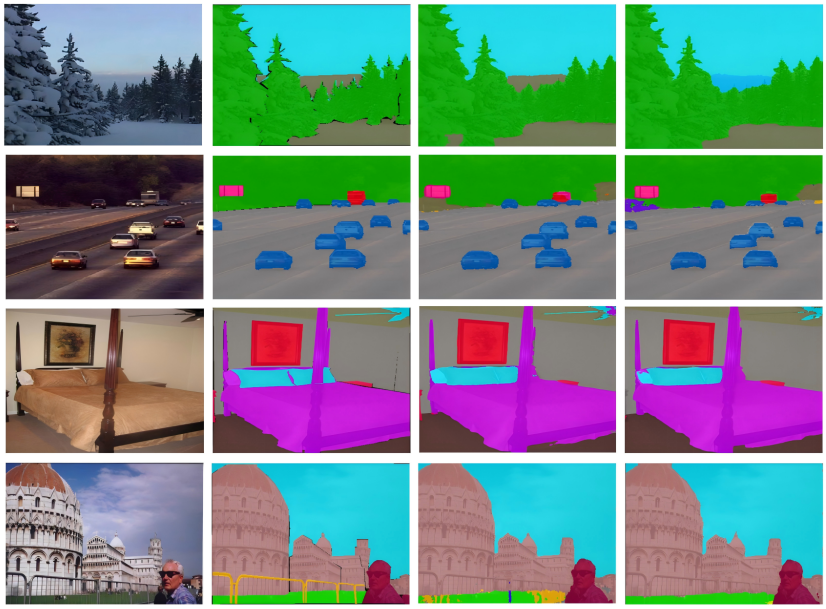

Figure 7: Results on additional segmentation samples with LRF-Dyn and LRF-SSA

Additionally, we visualized the qualitative results of our model on the ADE20K dataset. As shown in Fig.7, our segmentation results exhibit remarkable effectiveness, with precise boundary delineation and enhanced semantic coherence across diverse scene contexts.

H Visualization of Effective Receptive Fields

We compare the Effective Receptive Field (ERF) (Luo et al., 2016) of the center pixel on popular backbone networks before and after training, as shown in Fig. 8. The ERF distribution represents the contributions of every pixel on the input space to the central pixel in the final output feature maps. For visualization, we randomly select 50 images from the ImageNet-1K validation set, resize them to a resolution of 224×224, and then calculate the ERF values with the auto-grad mechanism.

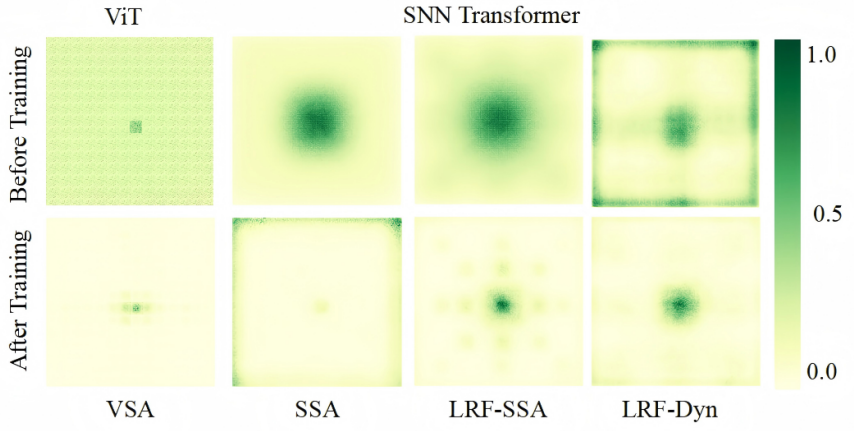

Figure 8: Comparison of Effective Receptive Field (ERF) among various architectures.

Before training, both the SSA method and the LRF-SSA method exhibit similar local receptive fields due to the presence of attention mechanisms. However, as training progresses, SSA gradually develops an almost global receptive field, whereas LRF-SSA retains a more local property. This behavior can be attributed to the LRF module, which enables the model to capture richer local information. In addition, LRF-Dyn also demonstrates a certain degree of local modeling ability, which contributes to improving the overall performance of the model. The effective receptive field visualizations further validate the effectiveness of our proposed approach.