BIO-INSPIRED NEURON-LIKE ENCODING AND ARTIFICIAL PHOTORECEPTOR LAYERS FOR EFFICIENT SPIKING NEURAL NETWORKS

CHING-TENG HEISH[1], YUANG-KAI WANG[2]

[12] Department of Electrical Engineering, Fu Jen Catholic University, New Taipei City 242062, Taiwan
E-MAIL: 412216035@m365.fju.edu.tw, ykwang@fju.edu.tw

Abstract:

Recent advancements in hardware technologies have significantly accelerated the development of artificial intelligence, deep learning, and neuromorphic computing. However, the unique characteristics of spike-based data constrain the performance and practicality of spiking neural networks (SNNs), often rendering them less competitive than convolutional neural networks (CNNs). While some approaches improve SNN performance by leveraging non-spiking data for training, they deviate from the core principles of neuromorphic computing. In this study, we propose a biologically inspired spike encoding framework based on neuron-like signal generation. Specifically, we introduce an artificial layer of rods and cones to simulate the functionalities of retinal photoreceptors, enabling the encoding of both color and luminance information into spike signals. This design offers a more comprehensive visual representation for SNNs. Moreover, we revisit the image acquisition pipeline and propose the concept of photon-based data, emphasizing the critical role of temporal resolution in static image encoding and spike signal formation. Experimental results validate that our neuron-based spike encoding and artificial photoreceptor layer significantly enhance spike information density, leading to improved SNN performance. Our findings aim to address key limitations in neuromorphic vision systems and contribute to expanding the applicability of SNNs across real-world domains.

Keywords:

Neuron-like Encoding; Artificial Layer of Rods and Cones; Photon Data; Neural Encoding; Neuromorphic Computing

1. Introduction

With the rapid advancement of technology, particularly the enhancement of GPU performance, the application of General Purpose computing on Graphics Processing Units (GPGPU) has been significantly propelled. As GPGPU technology has evolved, research in artificial intelligence (AI) has shifted from machine learning (ML) toward deep learning (DL), leveraging large-scale data to train artificial neural networks (ANN) for tasks such as automated classification and detection.

The progress in DL has led to numerous powerful applications; however, it also entails high energy consumption. To address this issue, scientists have introduced neuromorphic computing, which employs spiking neural networks (SNN) to reduce power consumption. Nevertheless, event data, which only captures dynamic texture variations and lacks color information, imposes limitations on its applicability. For instance, when a robot utilizes an event camera, it may face challenges in decision-making due to its inability to recognize colors or acquire texture information when stationary.

The image acquisition device in a computer vision system functions as the "eyes" of a computer, responsible for capturing light and converting it into electronic signals. Modern conventional cameras predominantly utilize CMOS sensors, whose core components consist of MOSFETs composed of NMOS and PMOS transistors. This configuration endows conventional cameras with low-power consumption characteristics. During the exposure period, light passes through the shutter, aperture, and Bayer filter before reaching the photodiode, where it is converted from optical energy into electrical energy. After simple processing, the data is stored as a static image. [1]

Recently, event cameras, inspired by the human retinal mechanism, have emerged as a novel alternative. These cameras use photodiodes, capacitors, and comparators to record luminance changes in real time, generating event data consisting of spatial coordinates (x, y), temporal information (t), and polarity (p). This architecture enables event data to achieve high dynamic range and spatiotemporal resolution while mitigating motion blur and reducing power consumption.

However, the inability to capture color information remains a significant limitation, restricting its applications. [2]

The development of machine vision is increasingly inspired by biological principles, with neuromorphic computing emerging as a key research area aimed at reducing energy consumption. Early models such as the Hodgkin-Huxley model provided precise representations of neural dynamics but were difficult to implement in practice. Consequently, the Leaky Integrate-and-Fire (LIF) model became the foundation for SNNs. However, due to the discrete nature of spike-based signals, SNNs face challenges in utilizing gradient descent for weight optimization, leading to the Dead Neuron Problem [3, 4]. To address this issue, researchers have proposed ANN-to-SNN conversion methods, modifying Convolutional Neural Network (CNN) weights and integrating the Integrate-and-Fire (IF) mechanism to transform CNNs into SNNs. While this approach enhances performance, it requires extended computational time. Alternative solutions, such as SuperSpike [5] and SLAYER [6], approximate spike functions with steep curves, making them differentiable for weight updates. SLAYER further refines this approach by incorporating time-dependent error distribution strategies.

However, methods like SLAYER cannot be easily applied to other SNN models. The development of Surrogate Gradient Descent [7] techniques has enabled weight updates through differentiable approximations, effectively mitigating the dead neuron problem. This advancement has driven the evolution of SNN architectures such as DECOLLE[8], S-ResNet [9], SEW-ResNet [10] and SpikFormer [11]. DECOLLE introduces a local learning strategy to reduce memory consumption, while S-ResNet pioneers the integration of ResNet structures into SNNs, albeit with training instability. SEW-ResNet resolves residual learning challenges with its Spike-Element-Wise (SEW) module, enabling the successful training of deep SNNs with over 100 layers. Meanwhile, SpikFormer incorporates a Transformer-based architecture, utilizing Spiking Self-Attention (SSA) to convert floating-point values into spike representations, thereby reducing computational complexity and power consumption through additive operations.

The encoding methods used in SNNs are diverse. Early research primarily relied on spike-based neural coding, such as Rate Coding and Temporal Coding. Rate Coding encodes information based on spike frequency, with common variants including Count Rate Coding, Density Rate Coding, and Population Rate Coding. In contrast, Temporal Coding focuses on spike timing, also known as Time-to-First-Spike. [3, 4, 12]

Currently, numerous studies employ event data for training SNNs, using datasets such as CIFAR10-DVS[13] and DVS128-Gesture[14], all captured by event cameras. However, event cameras are expensive and incapable of recording color information. Some studies at-tempt to simulate event data from conventional video sequences using methods such as PIX2NVS[15] and ESIM[16]. Nevertheless, the lack of color information continues to pose a significant challenge.

To enhance SNN performance, recent studies [10, 11, 17, 18] have explored training SNNs directly with color static images, converting them into spike-based representations using Spike Encoding Layers. Experimental results indicate that this approach significantly improves SNN performance, bringing it closer to CNNs. However, the repeated input of static images differs from biological vision, and the way Spike Encoding Layers process static images does not fully align with the mechanisms of biological visual neurons. Thus, we argue that non-spike-based signals, such as color static images, should not be used in training SNNs through repetitive input.

From the perspective of the human visual system, light signals are processed by rod and cone cells in the retina before being relayed by bipolar and ganglion cells to the brain for recognition [19]. However, current SNN training methods still differ from biological vision, and bridging this gap remains a crucial direction for future research.

Therefore, this study examines existing spike-based neural encoding methods and proposes a neuron-like encoding scheme inspired by retinal structures. This approach converts static images into spike data while preserving temporal and frequency information. Additionally, by designing an artificial layer of rods and cones, this method integrates color and brightness information into spike data in a multi-dimensional manner, enabling SNNs to receive near-complete visual signals.

Furthermore, we conduct experiments to analyze the impact of neuron-like encoding and the artificial layer of rods and cones on SNN performance and evaluate the advantages and limitations of the proposed encoding scheme. By examining the image acquisition process, we highlight the importance of temporal resolution in data representation, leading to the conceptualization of photon data. Through experiments, we investigate the relationship between temporal resolution and static images, as well as its influence on SNN performance, aiming to understand how static images contribute to enhancing SNN efficiency.

2. Method

In recent developments, SNNs have predominantly utilized static images as input. However, neuromorphic computing aims for SNNs to process spike signals rather than static images. Although Rate Coding and Temporal Coding can convert images into Poisson Spikes, these methods often

result in incomplete information encoding. Therefore, we explore a more biologically plausible approach to spike generation.

Drawing inspiration from the human visual and nervous systems, we consider how rod and cone cells in sensory neurons receive light stimuli and convert them into spike signals. These cells themselves do not perform filtering or real-time information processing; instead, subsequent neurons integrate the signals. Based on this observation, we argue that spike signals should be generated by neurons and that the conversion process should not interfere with one another.

The LIF model is a widely used neuron model in neuromorphic computing. However, in modern discrete systems with low temporal resolution, the use of the LIF model often results in excessively long time steps, significantly increasing computational cost and processing time. To address this issue, we adopt the IF model, which omits the membrane potential decay characteristic, allowing for more stable spike generation over shorter time intervals. This approach, where neuron models are used to encode spike signals, is referred to as Neuron Spike Encoding. Specifically, encoding with the LIF model is termed LIF Coding, whereas encoding with the IF model is referred to as IF Coding.

The process of energy accumulation in IF Coding can be described by (1), where $E_{mem}(t)$ represents the membrane potential at time t, and $E_L(t)$ represents the external light energy at time t. The neuron's spike signal generation is determined by (2), where $S(t)$ denotes the spike signal at time t, and θ represents the firing threshold. A spike is triggered if the membrane potential at time t, exceeds the firing threshold; otherwise, no spike is generated. Furthermore, when a neuron fires a spike, a certain amount of membrane potential energy must be converted into the spike signal, reducing the membrane potential. This reduction is represented in (3), where the membrane potential is decreased by one firing threshold unit upon spike generation.

Regarding the image acquisition process, we assume that the external light source corresponds to a static image. During the exposure period, energy accumulates while being subject to minor fluctuations due to thermal noise and internal circuit noise. This process is described by (4), where T is the exposure time, $P(T)$ represents the pixel value obtained after exposure, and $E_L(t)$ and $E_N(t)$ represent the external light energy and noise energy, respectively, over continuous time T. However, since noise typically does not significantly affect image formation, we further assume an idealized scenario where the imaging process is noise-free. This allows (4) to be simplified into (5), where the average accumulated light energy over the exposure period, E_{L_avg}, is equal to the pixel value $P(T)$.

Next, considering the 8-bit unsigned integer image storage format, the light energy values range from 0 to 255 and are often normalized to a 0–1 scale before processing. This normalization is defined in (6), where $P(T)$ represents the pixel value, MAX_I is the maximum pixel intensity (255 for uint8 images), and E_{L_norm} represents the normalized pixel value.

Building upon (5) and (6), we assume 256 discrete sampling time steps within the exposure period T, setting the initial membrane potential to 0 and the firing threshold to 1 in the IF model. Under these conditions, spikes start occurring at time step 2 and continue until time step 256. The accumulated spike count reconstructs the image, and the process resets at time step 257. Thus, the IF model can generate spike signals that accurately represent static images within 256 discrete time steps.

$$E_{mem}(t) = E_{mem}(t-1) + E_L(t) \tag{1}$$

$$S(t) = \begin{cases} 1, E_{mem}(T) > \theta \\ 0, elsewise \end{cases} \tag{2}$$

$$E_{mem}(t) = E_{mem}(t) - S(t) \times \theta \tag{3}$$

$$P(T) = \frac{\int E_L(t) + E_N(t)}{T} \tag{4}$$

$$P(T) = E_{L\_avg} = \frac{\int E_L(t)}{T} \tag{5}$$

$$0 \le \frac{P(T)}{MAX_I} = E_{L\_norm} \le 1 \tag{6}$$

The human retina primarily consists of cone cells and rod cells, whose spike signals collectively form visual perception. The absence of either type of information can significantly affect visual processing. Static color images preserve both chromatic and luminance information to the greatest extent. Therefore, we propose an Artificial Layer of Rods and Cones, which converts visual information into spike signals through artificial cone and rod cells. These spike signals are then provided to a SNN for training, enabling the network to learn comprehensive visual representations.

Artificial cone cells are designed based on the functional characteristics of biological cone cells. Using a neuron model, they encode color images into spike signals, with separate spike encoding applied to the red, green, and blue (RGB) channels. In contrast, artificial rod cells are responsible for encoding global luminance into spike signals, simulating retinal brightness perception. To achieve this, the luminance channel L is incorporated, as expressed in (7). Here, S_C represents the generated spike signal, En denotes the encoding method, C refers to the color channel, and p_c is the pixel intensity of a given channel. This approach aims to maximize the simulation of the visual information provided by sensory neurons in the retina.

$$S_C = En(p_c), C = R, G, B, L \tag{7}$$

Based on the image acquisition process of a traditional camera, light energy is converted into electrical energy and accumulates in a capacitor during the exposure period, with partial loss due to the photoelectric effect. The pixel value is determined by the accumulated charge. The exposure time can be discretized into X time steps, where the total exposure time for a static image is $X \times t$ seconds, as expressed in (8), where T represents the total exposure duration, X is the number of discrete time steps, and t is the sampling interval. Assuming that the charge increases at each discrete time step, the accumulated electrical energy can be expressed as (9), where Q_{total} denotes the total accumulated charge in the capacitor, $Q[n]$ represents the charge at each sampling instance, and n refers to the discrete time step.

Assuming that the photoelectric effect requires a sufficient amount of photon energy to strike the photosensitive element, we consider an incoming photon flux composed of N photons. If the photon energy is insufficient, no photons are assumed to be detected, as described in (10), where n represents the discrete time step, and $N_p[n]$ and $E_L[n]$ denote the number of photons and the total light energy at that moment, respectively. The occurrence of the photoelectric effect inevitably leads to energy loss, which is formulated in (11), where E_L is the total light energy, a is the energy loss constant, and Q represents the generated electrical charge.

If the temporal resolution is increased to the point of capturing the instantaneous occurrence of the photoelectric effect, the process can be considered as a discrete event. Thus, (10) can be rewritten as (12), where N photons are regarded as a light energy spike signal. Here, n represents the discrete time step, and $S[n]$ and $E_L[n]$ denote the spike signal and total light energy at that moment, respectively. Furthermore, if we define the minimum energy required for the photoelectric effect as the spike energy, we obtain (13), where E_s is the energy per spike event, N is the minimum number of photons required to trigger the photoelectric effect, and E_p is the energy per photon. Under this temporal resolution, the total light energy at time n directly influences whether a spike signal occurs, as described in (14). By substituting (11) and (14) into (9) and reorganizing, we derive (15), where S_{sum} represents the total number of spikes, $S_p[n]$ denotes the spike signal at time n, and X represents the number of discrete time steps. This formulation closely resembles Count Rate Coding, in which the average number of spikes occurring during the exposure period is given by (16). Here, S_{avg} represents the average spike occurrence during exposure, aligning with the formula used in Count Rate Coding.

The accumulation of photon spike signals reduces temporal resolution, effectively functioning as a downsampling operation along the time dimension. Conversely, Count Rate Coding simulates spike generation by increasing the number of discrete time steps, thereby enhancing temporal resolution, effectively serving as upsampling along the time dimension.

Based on this framework, we define the spike signals obtained from the conversion of static images as Photon Data. However, when the photon energy is insufficient to induce the photoelectric effect, energy loss still occurs, resembling the analog-to-digital conversion error in traditional cameras. This results in numerical discrepancies within the Photon Data.

$$T = X \times t \tag{8}$$

$$Q_{total} = \sum_{n=0}^{X} Q[n] \tag{9}$$

$$N_p[n] = \begin{cases} N, if\ E_L[n]\ is\ large\ enough \\ 0,\ elsewise \end{cases} \tag{10}$$

$$E_L = a \times Q \tag{11}$$

$$S[n] = \begin{cases} 1, if\ E_L[n]\ is\ large\ enough \\ 0,\ elsewise \end{cases} \tag{12}$$

$$E_s = N \times E_p \tag{13}$$

$$E_L[n] = S[n] \times E_s \tag{14}$$

$$S_{sum} = \sum_{n=0}^{X} S_p[n] \tag{15}$$

$$S_{avg} = \frac{S_{sum}}{X} \tag{16}$$

3. Experiment

Since we consider IF Coding to be an ideal spike encoding method, we first compare the spike characteristics of IF Coding, Rate Coding, and Temporal Coding. Assuming five normalized pixel values of 0, 0.25, 0.5, 0.75, and 1, the results are shown in Figure 1. We observe that while Rate Coding generates a large number of spikes, the random triggering mechanism results in a lack of temporal relationships between spikes, making it difficult to reconstruct the original values accurately. In contrast, Temporal Coding produces spikes with a clear temporal structure that reflects signal intensity, but the overall number of spikes is extremely low. IF Coding, however, not only accurately reconstructs the original values but also preserves the temporal relationship corresponding to signal strength.

To validate that IF Coding conveys more information, we applied IF Coding to convert CIFAR-10 [20] images into grayscale and color spike signals and then trained an SNN for

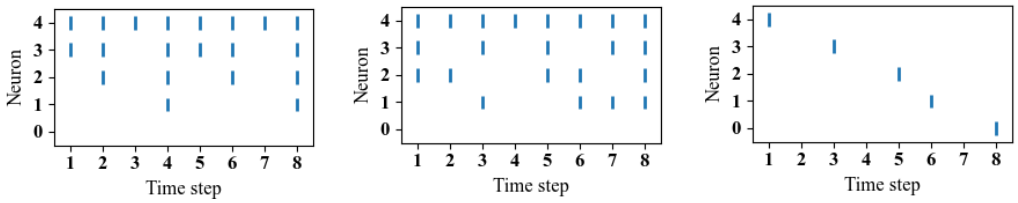

FIGURE 1. Comparison of Visualized Spike with Different Encoding method. Neuron numbers 0 to 4 correspond to the pixel values 0, 0.25, 0.5, 0.75, and 1, respectively. (a) Integrate-and-Fire Coding. (b) Rate Coding. (c) Temporal Coding.

testing and comparison. The results, presented in Table 1, show that across four different SNN models, training with IF Coding consistently achieved higher accuracy during testing compared to training with Rate Coding and Temporal Coding. This confirms that IF Coding carries richer information, enabling SNNs to learn more comprehensive data features. Additionally, we found that incorporating grayscale images had minimal impact on accuracy, potentially due to the distinct functional characteristics of cone and rod cells. Furthermore, we compared the performance of IF Coding with the Spike Encoding Layer for data conversion. The results indicate that for SEW-ResNet, both methods achieve comparable accuracy, whereas for SpikFormer, the Spike Encoding Layer demonstrates a slight advantage. However, in terms of temporal representation, IF Coding effectively observes the image only once within 256 time steps, whereas the Spike Encoding Layer continuously processes the full image at each time step, leading to an n-fold difference in temporal scale. Despite this difference in temporal resolution, IF Coding still performs competitively and surpasses conventional encoding methods, highlighting the potential of spike encoding based on neural operational principles as a promising direction for further development.

Next, we can simulate the generation and acquisition of Photon Data and static images through temporal upsampling and downsampling. Based on probability theory, when the number of discrete time points used for simulation is sufficiently large, the statistical average of the simulated spike signals converges toward the spike firing probability.

To determine the required number of time points, we employ Count Rate Coding for simulation and compare the Peak Signal-to-Noise Ratio (PSNR) variations of static images under single-precision floating-point and 8-bit unsigned integer formats. Using CIFAR-10 images for testing, we compute the average PSNR values for both the training and test datasets. As shown in Figure 2, the PSNR curve follows a log-linear relationship, where each tenfold increase in discrete time points results in an approximate tenfold increase in PSNR. When downsampling reaches 1,000 time points, the PSNR slope for the 8-bit format begins to flatten, and by 100,000 time points, it approaches zero.

TABLE 1. Comparison of Accuracy for SNN Training with Different Encoding Methods and Data Combinations

Model	Accuracy (%)	Coding	Data	Time step
SLAYER	47.99	Rate	RGB	256
	48.33	Temporal		
	54.14	IF		
	50.10	Rate		
	52.95	Temporal		
	57.66	IF		
	51.09	Rate		
	54.37	Temporal		
	57.38	IF		
DECOLLE	60.15	Rate	RGBL	256
	42.10	Temporal		
	64.97	IF		
SEW-Resnet	80.29	Rate		
	63.05	Temporal		
	83.46	IF		
SpikFormer	88.70	Rate		
	74.41	Temporal		
	92.21	IF		
SEW-Resnet	83.42	Encoding Layer	Static RGB Image	6
SpikFormer	93.34/ *95.19			4

*The data is sourced from the original research paper.

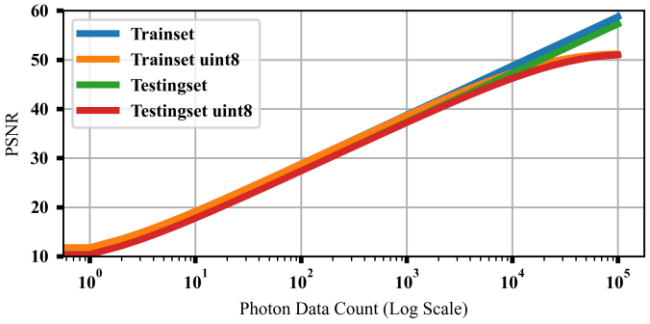

FIGURE 2. PSNR Curve Comparing Downsampled Photon Data-Based Static Images and Original Static Images. Although PSNR values vary depending on the images, the overall log-linear trend remains consistent. Here, "Trainset" and "Testingset" refer to the training set and testing set of the CIFAR-10 dataset, respectively. The suffix "uint8" indicates that the images are first converted to the uint8 format before being compared with the original images.

In the end, we examine the impact of temporal resolution on SNN performance. We generate static images with varying resolutions by using 1, 10, 100, 1,000, 10,000, and 100,000 sets of Photon Data as training inputs for the SNN. The

corresponding SNN accuracy results are presented in Figure 3. As temporal resolution increases, SNN accuracy also improves , reaching its peak when each image is generated from more than 10,000 sets of Photon Data. This indicates that SNN performance is highly dependent on temporal resolution, and achieving optimal results requires data with sufficiently high temporal resolution.

4. Conclusion

In alignment with the principles of neuromorphic computing, we enhance SNN learning efficiency by maximizing the information capacity carried by spike signals through Neuron Spike Encoding and the Artificial Layer of Rods and Cones. Additionally, we identify the critical role of temporal resolution in various data types, including static images, spike data, and photon data, and its significant impact on SNN performance. We hope this study will drive further advancements, enabling SNNs to overcome current limitations and expand their application domains.

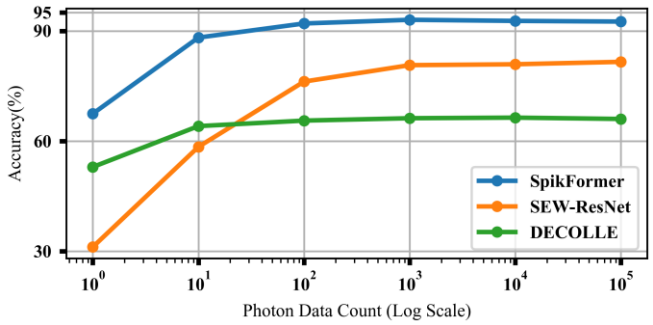

FIGURE 3. Accuracy Comparison Curve of Static Images with Different Temporal Resolutions on the CIFAR-10 Test Dataset. As the photon data count increases, the resulting accuracy improves accordingly. However, when each static image is obtained through downsampling from 1,000 photon data counts, the accuracy growth begins to plateau. Due to its inherent characteristics, the SLAYER model failed to converge and was therefore not utilized in this experiment.

Reference

[1] A. El Gamal and H. Eltoukhy, "CMOS image sensors," *IEEE Circuits and Devices Magazine,* vol. 21, no. 3, pp. 6-20, 2005.

[2] G. Gallego *et al.*, "Event-Based Vision: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 44, no. 1, pp. 154-180, 2022.

[3] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge: Cambridge University Press, 2002.

[4] J. K. Eshraghian *et al.*, "Training Spiking Neural Networks Using Lessons From Deep Learning," *Proceedings of the IEEE,* vol. 111, no. 9, pp. 1016-1054, 2023.

[5] F. Zenke and S. Ganguli, "SuperSpike: Supervised Learning in Multilayer Spiking Neural Networks," *Neural Computation,* vol. 30, no. 6, pp. 1514-1541, 2018.

[6] S. B. Shrestha and G. Orchard, "Slayer: Spike layer error reassignment in time," *Advances in neural information processing systems,* vol. 31, 2018.

[7] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate Gradient Learning in Spiking Neural Networks: Bringing the Power of Gradient-Based Optimization to Spiking Neural Networks," *IEEE Signal Processing Magazine,* vol. 36, no. 6, pp. 51-63, 2019.

[8] J. Kaiser, H. Mostafa, and E. Neftci, "Synaptic Plasticity Dynamics for Deep Continuous Local Learning (DECOLLE)," *Frontiers in Neuroscience,* vol. 14, 2020.

[9] Y. Hu, H. Tang, and G. Pan, "Spiking Deep Residual Networks," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 34, no. 8, pp. 5200-5205, 2023.

[10] W. Fang, Z. Yu, Y. Chen, T. Huang, T. Masquelier, and Y. Tian, "Deep residual learning in spiking neural networks," *Advances in Neural Information Processing Systems,* vol. 34, pp. 21056-21069, 2021.

[11] Z. Zhou *et al.*, "Spikformer: When Spiking Neural Network Meets Transformer," in *The Eleventh International Conference on Learning Representations*, 2023.

[12] W. Guo, M. E. Fouda, A. M. Eltawil, and K. N. Salama, "Neural Coding in Spiking Neural Networks: A Comparative Study for Robust Neuromorphic Systems," *Frontiers in Neuroscience,* vol. 15, 2021.

[13] H. Li, H. Liu, X. Ji, G. Li, and L. Shi, "CIFAR10-DVS: An Event-Stream Dataset for Object Classification," *Frontiers in Neuroscience,* vol. 11, 2017.

[14] A. Amir *et al.*, "A Low Power, Fully Event-Based Gesture Recognition System," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7388-7397.

[15] Y. Bi and Y. Andreopoulos, "PIX2NVS: Parameterized conversion of pixel-domain video frames to neuromorphic vision streams," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 1990-1994.

[16] H. Rebecq, D. Gehrig, and D. Scaramuzza, "ESIM: an Open Event Camera Simulator," *Conf. on Robotics Learning (CoRL),* 2018.

[17] M. Yao *et al.*, "Attention Spiking Neural Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 45, no. 8, pp. 9393-9410, 2023.

[18] L. Deng *et al.*, "Rethinking the performance comparison between SNNS and ANNS," *Neural Networks,* vol. 121, pp. 294-307, 2020.

[19] J. V. Forrester, A. D. Dick, P. G. McMenamin, F. Roberts, and E. Pearlman, *The Eye*, Fourth Edition ed. W.B. Saunders, 2016.

[20] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009.