

SpikeSen: Low-Latency In-Sensor-Intelligence Design With Neuromorphic Spiking Neurons

Ziru Li¹, Graduate Student Member, IEEE, Qilin Zheng, Graduate Student Member, IEEE,
Yiran Chen¹, Fellow, IEEE, and Hai Li¹, Fellow, IEEE

Abstract—In-sensor-processing (ISP) paradigm has been exploited in state-of-the-art vision system designs to pave the way towards power-efficient sensing and processing. The redundant data transmission between sensors and processors is significantly minimized by local computation within each pixel. However, existing ISP designs suffer from limited frame rates and degraded fill factors. In this brief, we introduce a low-latency in-sensor-intelligence neuromorphic vision system using neuromorphic spiking neurons, namely **SpikeSen**. **SpikeSen** directly operates on the photocurrents and executes the computation in the frequency domain, reducing the long exposure time and speeding up the computation. Experiments show that **SpikeSen** can achieve more than 6.1× computation speedup compared to existing ISP designs with competitive energy consumption per pixel.

Index Terms—In-sensor-processing, neuromorphic computing, low latency, frequency-domain computation, CMOS.

I. INTRODUCTION

THE PROLIFERATING edge computing techniques greatly reduce the energy consumption and latency of data transmission by moving computation resources close to source data in low-power real-time applications such as Internet-of-Things (IoT), portable devices, robots, etc. Vision systems that incorporate image sensors and processors are critical components in edge computing devices. The limited power budget and latency requirement have triggered a series of near-sensor-processing (NSP) vision system designs for image processing applications [3], [5], [7], [15]. These designs place low-power processors near the image sensor on a single chip to reduce the physical distance between the raw image data and the processing elements. Nevertheless, the power-hungry analog-to-digital converters (ADCs) between sensors and processors still hinder the vision systems from pursuing higher energy efficiency and performance.

In-sensor-processing (ISP) paradigm recently gains attention owing to its capability of building ultra-low-power vision

systems targeted on always-on image processing applications. In the ISP designs [8], [9], [12], [13], [14], processing elements are integrated within the pixels of the CMOS sensors. The processing element directly fetches the analog outputs from sensor pixels and performs image pre-processing computation locally, such as multi-layer perceptrons (MLPs), support vector machines (SVMs), or the 1st layer of binary neural networks (BNNs). This design concept reduces the considerable power of analog-to-digital conversion, saves memory access cost and bandwidth between sensors and processors, and avoids the need for large buffers for raw images. Nevertheless, unresolved drawbacks still exist in prior ISP designs, including the degraded fill factor and the long exposure time. Advances in neuromorphic engineering, inspired by the computation mechanisms of human brains, have led to a generation of large-scale spike-based processors for cognitive computing. Representative neuromorphic designs [1], [2], [4] perform spike-based computation in which the data are represented by the temporal information of spike trains. With simple synapse and neuron circuit implementations, neuromorphic computing hardware becomes a promising solution to pursuing higher power efficiency. Conventional spike-based image sensors [10], [11] transforms the photocurrent values to the spike frequency or timing in each pixel circuit, but they fail to implement computation associating multiple pixels. Developing a scheme which benefits from the direct integration of ISP and neuromorphic computing hardware becomes a valuable research topic.

To overcome the challenges in existing ISP vision systems, we propose a low-latency in-sensor-intelligence design with neuromorphic spiking neurons, namely **SpikeSen**. To our best knowledge, **SpikeSen** is the first ISP design that directly processes the photocurrents and computes the 1st layer of BNNs in the spike frequency domain. The main contributions in this brief include:

- 1) We propose a spike-based computing pixel (SCP) and SCP string circuit that leverages the CMOS-based neuromorphic spiking neuron and capacitive synaptic weight to process the photocurrent locally within each pixel and accumulate the partial results from adjacent pixels.
- 2) Based on the proposed SCP circuit substrate, we design **SpikeSen**, a low-latency ISP vision sensor with neuromorphic spiking neurons. **SpikeSen** executes convolution in the frequency domain with extremely low latency.
- 3) We optimize the mapping scheme and control flow of **SpikeSen** with a novel sub-SCP structure. The parallel operations of multiple sub-SCPs further simplify the control flow and speed up the computation.

Manuscript received 19 November 2022; revised 26 December 2022; accepted 7 January 2023. Date of publication 10 January 2023; date of current version 8 June 2023. This work was supported in part by the National Science Foundation under Grant 1910299 and Grant 1955246, and in part by the Army Research Office under Grant W911NF-19-2-0107. This brief was recommended by Associate Editor L. A. Camunas-Mesa. (Corresponding author: Ziru Li.)

The authors are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27707 USA (e-mail: ziruli@duke.edu; qilin.zheng@duke.edu; yiran.chen@duke.edu; hai.li@duke.edu).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TCSII.2023.3235888>.

Digital Object Identifier 10.1109/TCSII.2023.3235888

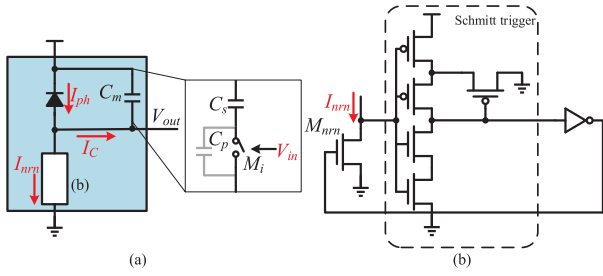


Fig. 1. (a) The structure of our proposed spike-based computing pixel (SCP). (b) The structure of the neuromorphic spiking neuron in SCP.

II. CHALLENGES IN IN-SENSOR-PROCESSING DESIGNS

There exist NSP designs [3], [5], [15] that deploy the processors close to the pixel array. ISP designs [9], [13], [14] go further by fusing the processing units with the pixels. Unresolved drawbacks exist in the prior CMOS-based ISP designs. First, the local analog processing element deployed within each pixel contains several memory cells to store model parameters, and thus the fill factor is degraded. For example, in [14], a register is added in each processing circuit to store weights. In [13], an SRAM macro that contains 16 SRAM cells is included in each in-pixel processing circuit. Second, the computation in the prior designs requires a long exposure time to accumulate photocurrents, which leads to a long computation latency. Third, the prior designs still require power-consuming ADCs to convert analog outputs to the digital domain.

III. NEUROMORPHIC VISION SENSOR DESIGN

In this section, we introduce SpikeSen from the basic circuit substrate to the top architecture. We start from the unit cell in SpikeSen named spike-based computing pixel (SCP) in Section III-A. Then we introduce how to build the interconnects between different SCPs in Section III-B. Based on the proposed circuit structure, we present our proposed spike-based neuromorphic vision sensor design in Section III-C.

A. Spike-Based Computing Pixel

The proposed SCP computes the product between the pixel value in the format of photocurrent and the weight value programmed to a capacitor. Fig. 1(a) depicts the structure of our proposed SCP. In each SCP, the “programmable capacitor” C_m is in parallel with the photodiode (PD) and in series with the spiking neuron. The spike neuron consists of a transistor M_{nrn} in parallel with a Schmitt trigger that controls the gate voltage of M_{nrn} . We adopt the Schmitt trigger designed in [6] due to its simplicity, as shown in Fig. 1(b).

Principle of SCP: Assume the current flowing through the spike neuron is I_{nrn} , the photocurrent is I_{ph} and the current charging C_m is I_C . One can derive the equations:

$$I_{nrn} + I_C = I_{ph}; I_C = C_m \frac{dV_{out}}{dt}. \quad (1)$$

Assume the high/low threshold voltages of the Schmitt trigger in the spiking neuron are V_{thH} and V_{thL} , respectively. When V_{out} increases from 0 and is lower than V_{thH} , M_{nrn} is cut off. I_{nrn} is much smaller than I_{ph} , leading to $I_C = I_{ph}$.

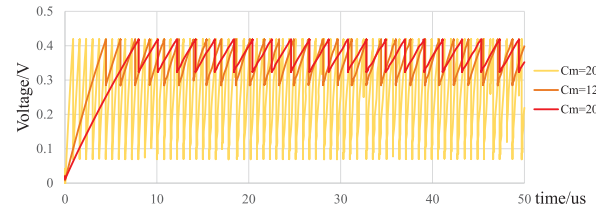


Fig. 2. Simulation V_{out} waveform of SCPs with different C_m values.

I_C keeps charging C_m until $V_{out} = V_{thH}$, toggling the output of the Schmitt trigger. M_{nrn} is then switched on with $I_{nrn} \gg I_{ph}$, leading to a large discharging current $I_C = -I_{nrn}$ that decreases V_{out} to V_{thL} instantly, based on Equation (1). Then M_{nrn} is turned off again. This charging/discharging loop generates the oscillating V_{out} , and its frequency is proportional to I_{ph} since the charging phase ($I_C = I_{ph}$) is dominant in the charging/discharging loop.

In an SCP, the output oscillating frequency is not only proportional to the input photocurrent I_{ph} , but also inversely proportional to the capacitance C_m since $\frac{dV_{out}}{dt} = \frac{I_C}{C_m}$ during the charging phase. As shown in Fig. 2, larger C_m leads to smaller V_{out} oscillation frequency. In our SCP design shown in Fig. 1(a), C_m consists of a transistor switch in series with a capacitor C_s and in parallel with a small capacitor C_p . The binary weight is represented by the gate voltage of the transistor switch M_i , denoted as V_{in} . When V_{in} is low and M_i is switched off, C_p leads to low capacitance state (LCS) of C_m . When M_i is switched on, C_p is bypassed and C_s leads to high capacitance state (HCS). By programming the weight to $\frac{1}{C_m}$, the output oscillating frequency of SCP will represent the product of the photocurrent strength and the weight.

B. SCP String for MAC Operation

The proposed SCP implements multiplication within each pixel unit in the frequency domain. Prior ISP designs [13], [14] require a long exposure time ($\sim 1ms$) to accumulate the photocurrents from multiple cells on the capacitors and convert the voltages to the digital domain as the computation results. Substituting the slow photocurrent accumulation with high-frequency oscillation greatly shortens the computation latency and boosts the performance. However, the connections between multiple pixels are also necessary to accumulate the multiplication results from different pixels.

To support multiply-and-accumulate (MAC) operations with our basic SCP design, we propose a structure called SCP string. Fig. 3(a) shows the structure of a 2-cell SCP string that consists of two basic SCP cells (SCP1 and SCP2) in series as a simplified example. SCP1 is connected to the power supply V_{DD} . The output is generated by the spiking neuron in SCP2. There is an additional initialization circuit including two bias resistors R_{bias} , which will be interpreted hereinafter. Since these two SCPs are connected in series, the following equations can be derived:

$$\begin{aligned} I_{nrn1} + I_{C1} &= I_{ph1}; I_{C1} = C_{m1} \frac{dV_{C1}}{dt}; \\ I_{nrn2} + I_{C2} &= I_{ph2}; I_{C2} = C_{m2} \frac{dV_{C2}}{dt}; \\ \frac{dV_{out}}{dt} &= \frac{dV_{C1}}{dt} + \frac{dV_{C2}}{dt}; \end{aligned} \quad (2)$$

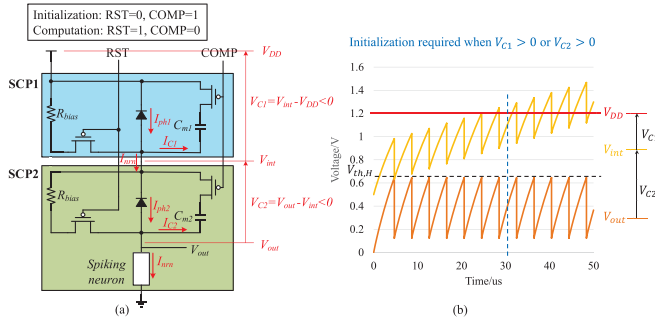


Fig. 3. (a) The structure of a 2-cell SCP string for MAC operation. (b) The simulation result of a 2-cell SCP string oscillating waveforms, including the output voltage V_{out} and the internal node voltage between the two cells V_{int} .

where V_{C1} and V_{C2} are the voltages across two SCPs. The series connection enables the oscillation to be modulated by both SCPs, i.e., $\frac{dV_{out}}{dt} = \frac{I_{C1}}{C_{m1}} + \frac{I_{C2}}{C_{m2}}$.

The oscillation behavior in the SCP string is similar to the oscillation in a single SCP. When V_{out} increases from 0V, I_{nm} is much smaller than I_{ph1} and I_{ph2} . Thus $I_{C1} = I_{ph1}$ and $I_{C2} = I_{ph2}$, leading to $\frac{dV_{out}}{dt} = \frac{I_{ph1}}{C_{m1}} + \frac{I_{ph2}}{C_{m2}}$. In this way, the increasing speed of V_{out} can represent the MAC result $\frac{I_{ph1}}{C_{m1}} + \frac{I_{ph2}}{C_{m2}}$. When V_{out} reaches the threshold voltage of the Schmitt trigger V_{thH} , I_{nm} becomes much larger than I_{ph1} and I_{ph2} , decreasing V_{out} instantaneously. The oscillation frequency is proportional to the MAC result between the input vector $[I_{ph1} \ I_{ph2}]$ and the weight vector $[\frac{1}{C_{m1}} \ \frac{1}{C_{m2}}]$. Following the same design principle, an n -cell SCP string that consists of n basic SCP cells in series can compute the MAC results of n -element input vectors and weight vectors. The SCP string enables the accumulation of the products calculated in multiple SCP cells in the frequency domain. The SCP cells are connected in series so that the voltage changes of all the SCPs are summed up. This overall voltage change is reflected by the change of output voltage V_{out} . Thus, the oscillation frequency of V_{out} represents the overall MAC result.

Weight representation and bias cancellation: As interpreted hereinbefore, the weight in each SCP is represented by $\frac{1}{C_m}$. However, for the implementation of binary neural networks, even if the -1 weights are represented by $\frac{1}{C_{m,HCS}}$ where $C_{m,HCS}$ is the high capacitance state value, they still positively contribute to the output oscillation frequency. To cancel this positive weight bias and enable ± 1 weights, an additional cycle is required to program all the capacitance values inversely. The final output should be the difference between the output frequency in the computation stage and the output frequency in this *bias cancellation* stage, i.e.,

$$\begin{aligned} \text{Output} &= f_{comp} - f_{bc} \\ &= \sum_{n=1}^N I_{ph,n} \frac{1}{C_{m,n}} - \sum_{n=1}^N I_{ph,n} \frac{1}{C'_{m,n}} \\ &= \sum_{n=1}^N I_{ph,n} \left(\frac{1}{C_{m,n}} - \frac{1}{C'_{m,n}} \right), \end{aligned} \quad (3)$$

where $C_{m,n}$ and $C'_{m,n}$ are C_m values in the n -th SCP in the SCP string during the computation stage and *bias cancellation* stage, respectively. $C'_{m,n}$ is $C_{m,HCS}$ while $C_{m,n}$ is $C_{m,LCS}$, or

vice versa. In this case, $+1$ weights are represented by $\frac{1}{C_{m,LCS}} - \frac{1}{C_{m,HCS}}$ and -1 weights are represented by $\frac{1}{C_{m,HCS}} - \frac{1}{C_{m,LCS}}$. Such a differential computing paradigm also contributes to better noise immunity.

The initialization of SCP string: The PDs should work in reverse bias. In an SCP string, we need to guarantee that the PD in each SCP is in reverse bias, which means $V_{C1} < V_{bias} < 0$ and $V_{C2} < V_{bias} < 0$. Hence, a bias resistor R_{bias} is added in parallel with PD and C_m in each SCP. The voltages of the internal nodes between adjacent SCPs are distributed evenly between V_{DD} and ground by the bias resistor string during the initialization stage, reversely biasing all the PDs. Fig. 3(b) shows the simulation result that manifests the oscillation behavior of the internal node voltage in a 2-cell SCP string in Fig. 3(a). Both the output voltage V_{out} and the internal node voltage V_{int} oscillate at the same frequency. It can be observed that V_{int} gradually approaches V_{DD} when the oscillation proceeds, leading to decreasing bias voltage of the PD in SCP1. To guarantee the reverse bias of PDs throughout the operation, a periodic initialization is required to prevent any two internal node voltages in the SCP string from intersecting each other. The initialization period depends on the values of the photocurrent I_{ph} and C_m , as well as the total voltage supply of the SCP string V_{DD} .

C. Spike-Based Neuromorphic Vision Sensor

1) Fundamental SpikeSen Architecture: The fundamental architecture overview of SpikeSen is shown in Fig. 4(a). To enable the formation of different SCP strings within SpikeSen, interconnects between adjacent SCP cells are necessary. Take 2×2 convolution as an example. A 4-cell SCP string executes the MAC result in a 2×2 window. We adopt a “Left-to-right column-by-column (LRCC)” interconnect scheme to form an SCP string with 4 SCPs, as shown by the red arrows in Fig. 4(b). In each 4-cell SCP string, the 4 SCP cells have 4 different interconnect modes and modulate the photocurrents with different weights. For example, the top-left SCP is connected to V_{DD} and the bottom-left SCP; the top-right SCP is connected to the bottom-right SCP and output the MAC results. In total, 8 controllable interconnects in each SCP are required to form the LRCC SCP string, as highlighted in Fig. 4(a). Note that when the adjacent SCPs are connected, the interconnects are merged. For example, the TL of SCP(x , $y+1$) is the same interconnect as the BR of SCP(x , y). Fig. 4(c) shows the ON/OFF state of the 8 interconnects under the 4 interconnect modes. Note that the 8 interconnects can support larger convolution kernels under our LRCC interconnect scheme. The difference lies in how to control the interconnects according to the position of each SCP in the SCP string.

To slide multiple kernels on the image and generate multiple output feature map channels, C binary weights are programmed to C_m in each SCP, where C is the output channel number. These C binary weights are at the same position of different kernels in a BNN layer. C output feature map channels are generated sequentially. Since for each output channel, the control flow is identical, we will concentrate on the single-channel convolution hereinafter.

2) Sub-SCP Structure Optimization: In the fundamental SpikeSen architecture, the entire SCP array is divided into

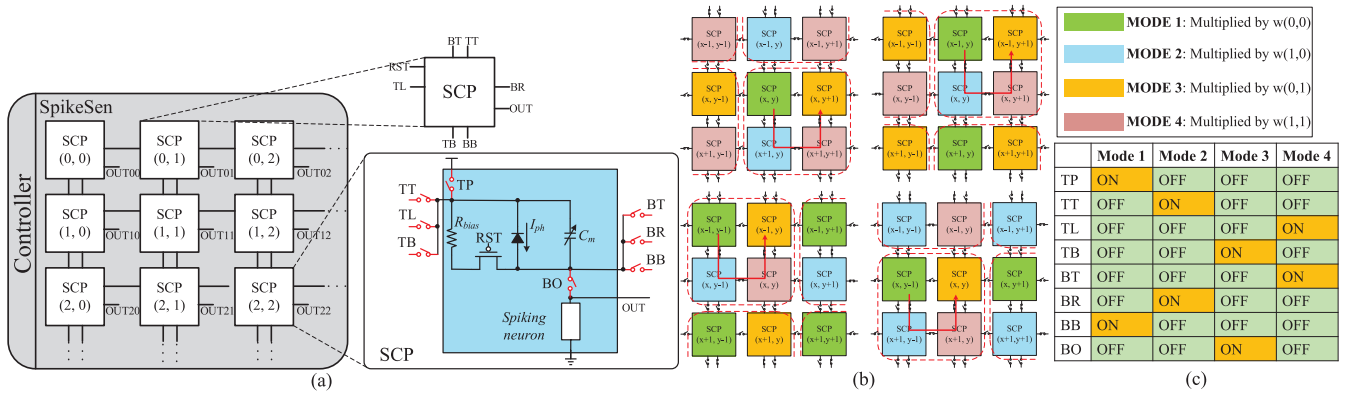


Fig. 4. (a) The overall architecture of SpikeSen. (b) The LRCC formation of multiple SCP strings in SpikeSen to perform a 2×2 convolution. (c) The ON/OFF state of the 8 interconnects in each SCP under the 4 interconnect modes.

multiple SCP strings that work in parallel with the same kernel. When one cycle of computation is completed, the interconnect mode of each SCP should be switched to form different SCP strings, which is equivalent to sliding the kernel over the image. Besides, the weights are also supposed to be shifted on the SCP array. The total cycle number of a single-channel convolution computation depends on the number of interconnect modes, e.g., 4 cycles for 2×2 convolution. Both the interconnect mode and the weight of each SCP need switching when one cycle of computation is completed.

The controllable interconnects using CMOS switches inject additional noises to the oscillating voltage output and degrade the computation accuracy. Therefore, we propose a novel sub-SCP structure of SCP that can implement the kernel sliding in a more efficient way than the fundamental SCP in Fig. 4(a). The proposed sub-SCP structure of an SCP is shown in Fig. 5. For a 2×2 convolution, the original SCP is divided into four sub-SCPs, and each sub-SCP contains two interconnects, one programmable C_m and one PD, respectively. All the four sub-SCPs in one SCP sense the same image region, and work simultaneously under one of the four interconnect modes respectively. The spiking neuron is added to the sub-SCP at the output side of the SCP string.

The benefits from the sub-SCP design are two-fold. First, since the interconnects are distributed to different sub-SCPs, no CMOS switches are needed to control the ON/OFF state of the interconnects. The noises caused by CMOS switches are avoided. Second, the weights in one convolution kernel are mapped to different sub-SCPs of a SCP. Since the sub-SCPs work in parallel under different interconnect modes, the computation of a single-channel convolution is completed in one cycle instead of multiple cycles in the fundamental architecture, thus reducing the total computation latency and boosting the throughput.

IV. EXPERIMENTS

A. Experiment Setup

We perform the post-layout simulation of the proposed circuits in Cadence Virtuoso with TSMC 65nm PDK. The size of the SCP array is set to 32×32 . In the evaluated structure, each SCP contains 4 sub-SCPs to perform a 2×2 convolution layer. Since the oscillation frequency of spiking neuron reaches above 2MHz , the computation stage is set to $5\mu\text{s}$ for

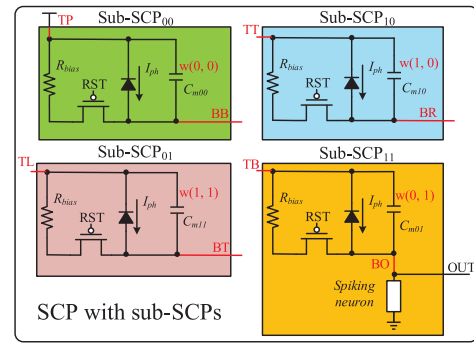


Fig. 5. The sub-SCP structure in each SCP.

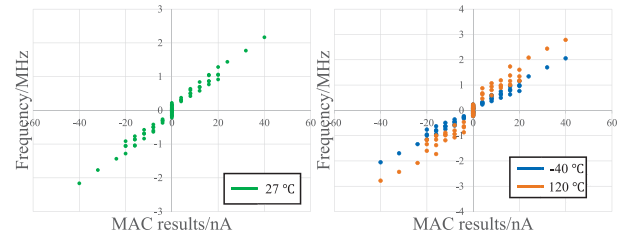


Fig. 6. The relationship between the final output of a 4-cell SCP string and the expected MAC results after *bias cancellation*.

each output channel, and the total computation latency of each output channel is $10\mu\text{s}$ including the *bias cancellation* stage. The initialization period is set to $5\mu\text{s}$.

B. Computational Accuracy Analysis

Fig. 6 shows the relationship between the oscillation frequency of a 4-cell SCP string and the expected MAC results after *bias cancellation*. Here we simulate with 4-cell SCP string to verify the computation results of a 2×2 convolution. In the simulation, LCS/HCS are set to $20\text{fF}/200\text{fF}$. The photocurrents range from $0 \sim 10\text{nA}$ with 6 levels. The initialization circuit is also included in the simulation. Process variations with a standard deviation of 10% are injected to C_m in each SCP. A $\pm 5\%$ voltage fluctuation is applied to the power supply of the SCP string. We perform Monte Carlo simulation with 160 data points under 27°C , 80 data points under -40°C and 80 data points under 120°C . The expected MAC results of 4 capacitive weights and 4 photocurrent-based inputs are

TABLE I
COMPARISON WITH EXISTING NSP AND ISP DESIGNS

Work	This work	2014 ISSCC [9]	2017 ISSCC [15]	2017 JSSC [3]	2020 DAC [14]	2021 TCASI [13]
Process	65nm	180nm	90nm/40nm	65nm	65nm	180nm
Supply voltage	2.5V/1V	3.3V/1.8V	3.3V/2.9V/1.8V/1.1V	2.5V/0.5~0.8V	1.2V	0.8~1.8V
Array size	32 × 32	64 × 64	1296 × 976	320 × 240	32 × 32	32 × 32
Pixel size	30μm × 30μm	28.8μm × 28.8μm	3.5μm × 3.5μm	7μm × 7μm	20μm × 20μm	35μm × 35μm
Fill factor	28.4%	18.32%	N/A	N/A	14%	9.14%
Processing mode	Analog Frequency-based	Mixed signal	Digital	Mixed signal	Analog Charge-based	Analog Charge-based
Latency	60us	N/A	N/A	N/A	368us	9.6ms
FoM (pJ/pixel-frame) ^a	22.06	14.1	66.5	73.7	13	~0.2

^aFoM = $P_{\text{pixel}} \times t_{\text{frame}}$, where P_{pixel} represents the average power of SCP and t_{frame} represents the latency of each frame (60us).

depicted by the X-axis. The final output frequency depicted by the Y-axis is linearly proportional to the expected MAC results, indicating the capability of the proposed SCP structure to perform MAC operations. The oscillation frequency of SpikeSen output increases with the temperature. The linearity and immunity to process variations are degraded under high temperature (120°C). The measured SNR under room temperature is 18.2dB.

C. Energy, Performance and Area Comparison

Table I shows the comparison between SpikeSen and existing NSP designs [3], [15] and ISP designs [9], [13], [14]. We scale the latency of each design according to a 2 × 2 convolution layer with 6 output channels. We select energy per frame per pixel as the Figure of Merit for fair energy efficiency comparison and scale the data to 65nm/1.2V node.

SpikeSen achieves at least 6.1× latency reduction over existing NSP designs [3], [15] and ISP designs [9], [13], [14]. SpikeSen consumes 22.06pJ pixel-wise energy per frame, which achieves at least 66.8% reduction compared to the digital and mixed-signal ISP counterparts [3], [15]. The extremely low computation latency of SpikeSen stems from the high-frequency-domain computation performed by SCPs. Because the oscillation is generated by charging and discharging the capacitor, SpikeSen also features minor static power dissipation during the oscillation. SpikeSen does not defeat [13], [14] in terms of energy per pixel. However, it should be noted that these designs generate binary MAC results by sensing the differential voltage outputs. To achieve competitive precision with SpikeSen, they require power-hungry ADCs to sense differential voltage as well as analog peripherals that assist the computation, leading to considerable power consumption and area overhead.

V. CONCLUSION

In this brief, we propose SpikeSen, a low-latency in-sensor-intelligence vision sensor design with neuromorphic spiking neurons. SpikeSen harnesses the novel spike-based computing pixel design and performs the computation in the frequency domain. SpikeSen significantly outperforms prior designs with more than 6.1× computation speedup and competitive energy consumption.

REFERENCES

- [1] F. Akopyan et al., "Truenorth: Design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 34, no. 10, pp. 1537–1557, Oct. 2015.
- [2] B. V. Benjamin et al., "Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations," *Proc. IEEE*, vol. 102, no. 5, pp. 699–716, May 2014.
- [3] K. Bong, S. Choi, C. Kim, D. Han, and H.-J. Yoo, "A low-power convolutional neural network face recognition processor and a CIS integrated with always-on face detector," *IEEE J. Solid-State Circuits*, vol. 53, no. 1, pp. 115–123, Jan. 2018.
- [4] M. Davies et al., "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, Jan./Feb. 2018.
- [5] Z. Du et al., "ShiDianNao: Shifting vision processing closer to the sensor," in *Proc. 42nd Annu. Int. Symp. Comput. Archit.*, 2015, pp. 92–104.
- [6] I. M. Filanovsky and H. Baltes, "CMOS Schmitt trigger design," *IEEE Trans. Circuits Syst. I, Fund. Theory Appl.*, vol. 41, no. 1, pp. 46–49, Jan. 1994.
- [7] R. LiKamWa, Y. Hou, J. Gao, M. Polansky, and L. Zhong, "RedEye: Analog ConvNet image sensor architecture for continuous vision with 2D material neural network image sensors," *ACM SIGARCH Comput. Archit. News*, vol. 44, no. 3, pp. 255–266, 2016.
- [8] L. Mennel, J. Symonowicz, S. Wachter, D. K. Polyushkin, A. J. Molina-Mendoza, and T. Mueller, "Ultrafast machine vision with 2D material neural network image sensors," *Nature*, vol. 579, no. 7797, pp. 62–66, 2020.
- [9] S. Park, J. Cho, K. Lee, and E. Yoon, "7.2 243.3 pJ/pixel bio-inspired time-stamp-based 2D optic flow sensor for artificial compound eyes," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, 2014, pp. 126–127.
- [10] C. Shoushun and A. Bermak, "Arbitrated time-to-first spike CMOS image sensor with on-chip histogram equalization," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 3, pp. 346–357, Mar. 2007.
- [11] X. Wang, W. Wong, and R. Hornsey, "A high dynamic range CMOS image sensor with inpixel light-to-frequency conversion," *IEEE Trans. Electron Devices*, vol. 53, no. 12, pp. 2988–2992, Dec. 2006.
- [12] C. Xu et al., "5.1 A stacked global-shutter CMOS imager with SC-type hybrid-GS pixel and self-knee point calibration single frame HDR and on-chip binarization algorithm for smart vision applications," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, 2019, pp. 94–96.
- [13] H. Xu et al., "Senputing: An ultra-low-power always-on vision perception chip featuring the deep fusion of sensing and computing," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 1, pp. 232–243, Jan. 2022.
- [14] H. Xu et al., "Utilizing direct photocurrent computation and 2D kernel scheduling to improve in-sensor-processing efficiency," in *Proc. 57th ACM/IEEE Design Autom. Conf. (DAC)*, 2020, pp. 1–6.
- [15] T. Yamazaki et al., "4.9 A 1ms high-speed vision chip with 3D-stacked 140GOPS column-parallel PES for spatio-temporal image processing," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, 2017, pp. 82–83.