

Event-Driven Tactile Learning with Location Spiking Neurons

Peng Kang*, Srutarshi Banerjee†, Henry Chopp†, Aggelos Katsaggelos*†, Oliver Cossairt*†

*Department of Computer Science, Northwestern University, Evanston, IL 60208, USA

†Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL 60208, USA

Email: {pengkang2022, SrutarshiBanerjee2022, henrychopp2017}@u.northwestern.edu,
{a-katsaggelos, oliver.cossairt}@northwestern.edu

Abstract—The sense of touch is essential for a variety of daily tasks. New advances in event-based tactile sensors and Spiking Neural Networks (SNNs) spur the research in event-driven tactile learning. However, SNN-enabled event-driven tactile learning is still in its infancy due to the limited representative abilities of existing spiking neurons and high spatio-temporal complexity in the data. In this paper, to improve the representative capabilities of existing spiking neurons, we propose a novel neuron model called “location spiking neuron”, which enables us to extract features of event-based data in a novel way. Moreover, based on the classical Time Spike Response Model (TSRM), we develop a specific location spiking neuron model – Location Spike Response Model (LSRM) that serves as a new building block of SNNs. Furthermore, we propose a hybrid model which combines an SNN with TSRM neurons and an SNN with LSRM neurons to capture the complex spatio-temporal dependencies in the data. Extensive experiments demonstrate the significant improvements of our models over other works on event-driven tactile learning and show the superior energy efficiency of our models and location spiking neurons, which may unlock their potential on neuromorphic hardware.^[2]

Index Terms—Spiking Neural Networks, spiking neuron models, location spiking neurons, event-driven tactile learning

I. INTRODUCTION

The tactile perception is a vital sensing modality that enables humans to gain perceptual judgment on the surrounding environment and conduct stable movement [1]. With the recent advances in material science and Artificial Neural Networks (ANNs), research on tactile perception begins to soar, including tactile object recognition [2]–[4], slip detection [5], and texture recognition [6], [7]. Unfortunately, although ANNs demonstrate promising performance on the tactile learning tasks, they are usually power-hungry compared to human brains that require far less energy to perform the tactile perception robustly [8], [9].

Inspired by biological systems, research on event-driven perception starts to gain momentum, and several asynchronous event-based sensors have been proposed, including event cameras [10] and event-based tactile sensors [11]. In contrast to

¹The TSRM is the classical SRM in the literature. We add the character “T” to highlight its difference with the LSRM.

²Copyright © 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works by sending a request to pubs-permissions@ieee.org.

standard synchronous sensors, such event-based sensors can achieve higher energy efficiency, better scalability, and lower latency. However, due to the high sparsity and complexity of event-driven data, learning with these sensors remains in its infancy [12]. Recently, several works [1], [11], [13] utilized Spiking Neural Networks (SNNs) [12], [14], [15] to tackle event-driven tactile learning. Unlike ANNs generally requiring expensive transformations from asynchronous discrete events to synchronous real-valued frames, SNNs can process event-based sensor data directly. Moreover, unlike ANNs employing artificial neurons [16]–[18] and conducting real-valued computation, SNNs adopt spiking neurons [19]–[21] and utilize binary 0-1 spikes to process information. This difference reduces the mathematical dot-product operations in ANNs to less computationally summation operations in SNNs. Due to the advantages of SNNs, these works are always energy-efficient and suitable for power-constrained devices. However, due to the limited representative abilities of current spiking neuron models and high spatio-temporal complexity in the event-based tactile data, these works still cannot sufficiently capture spatio-temporal dependencies and thus hinder the performance of event-driven tactile learning.

In this paper, to address the problems mentioned above, we make several contributions that advance event-driven tactile learning.

First, to enable richer representative abilities of existing spiking neurons, we propose a novel neuron model called “location spiking neuron”. Unlike existing spiking neuron models that update their membrane potentials based on time steps [22], location spiking neurons update their membrane potentials based on locations. Moreover, based on the Time Spike Response Model (TSRM) [19], we develop the “Location Spike Response Model”, henceforth referred to as “LSRM”. The TSRM is the classical SRM in the literature. We add the character “T (Time)” to highlight its difference with the LSRM. These location spiking neurons enable us to extract feature representations of event-based data in a novel way. Previously, SNNs adopted temporal recurrent neuronal dynamics to extract features from the event-based data. With location spiking neurons, we can build SNNs that employ spatial recurrent neuronal dynamics to extract features from the event-based data. We believe location spiking neuron models can have a broad impact on the SNN community

and spur the research on learning from event sensors like NeuTouch [11] or Dynamic Vision Sensors [10].

Next, we investigate the effectiveness of location spiking neurons and develop a hybrid model to capture the complex spatio-temporal dependencies in the event-driven data. The hybrid model combines an SNN with TSRM neurons and an SNN with LSRM neurons. Moreover, we introduce a location spike-count loss and a weighted spike-count loss to train the SNN with LSRM neurons and the hybrid model, respectively.

Last but not least, we apply our proposed models to event-driven tactile learning, including event-driven tactile object recognition and event-driven slip detection, and test them on three challenging datasets. Specifically, two sub-tasks are included in the task of event-driven tactile object recognition. The first sub-task requires models to determine the type of objects being handled. The second sub-task requires models to determine the type of containers being handled and the amount of liquid held within, which is more challenging than the first sub-task. In the task of event-driven slip detection, models need to accurately detect the rotational slip (“stable” or “rotate”) within 0.15s. Extensive experimental results demonstrate the significant improvements of our models over other state-of-the-art methods on event-driven tactile learning and show the superior energy efficiency of our models, which may bring new opportunities and unlock their potential on neuromorphic hardware.

To the best of our knowledge, this is the first work to propose location spiking neurons, introduce the LSRM, and build SNNs with location spiking neurons for event-driven tactile learning. The rest of the paper is organized as follows. In Section II we give an overview of related work on SNNs and event-driven tactile sensing and learning. In Section III we start by introducing notations for TSRM neurons and extending them to the specific location spiking neurons – LSRM neurons. We then propose models with LSRM neurons for event-driven tactile learning. Last, we provide implementation details and algorithms related to the proposed models. In Section IV we demonstrate the effectiveness and energy efficiency of our proposed models on different benchmark datasets. Finally, we discuss the broad impact of this work and conclude in Section V.

II. RELATED WORK

In the following, we give a brief overview of related work on SNNs and event-driven tactile sensing and learning.

A. Spiking Neural Networks (SNNs)

With the prevalence of Artificial Neural Networks (ANNs), computers today have demonstrated extraordinary abilities in many cognition tasks. However, ANNs only imitate brain structures in several ways, including vast connectivity and structural and functional organizational hierarchy [22]. The brain has more information processing mechanisms like the neuronal and synaptic functionality [23], [24]. Moreover, ANNs are much more energy-consuming than human brains. To integrate more brain-like characteristics and make artificial

intelligence models more energy-efficient, researchers propose Spiking Neural Networks (SNNs), which can be executed on power-efficient neuromorphic processors like TrueNorth [25] and Loihi [26]. Similar to ANNs, SNNs can adopt general network topologies like convolutional layers and fully-connected layers, but use different neuron models [21]. Commonly-used neuron models for SNNs are the Leaky Integrate-and-Fire (LIF) model [20] and the Time Spike Response Model (TSRM) [19]. Due to the non-differentiability of these spiking neuron models, it still remains challenging to train SNNs. Nevertheless, several solutions have been proposed, such as converting trained ANNs to SNNs [27], [28] and approximating the derivative of the spike function [29], [30]. In this work, we propose location spiking neurons to enhance the representative abilities of existing spiking neurons. These location spiking neurons maintain the spiking characteristic but employ the spatial recurrent neuronal dynamics, which enable us to build energy-efficient SNNs and extract features of event-based data in a novel way. Moreover, based on the optimization methods for SNNs with existing spiking neurons, we can derive the approximate backpropagation methods for SNNs with location spiking neurons.

B. Event-Driven Tactile Sensing and Learning

With the prevalence of material science and robotics, several tactile sensors have been developed, including non-event-based tactile sensors like the iCub RoboSkin [31] and the SynTouch BioTac [32] and event-driven tactile sensors like the NeuTouch [11] and the NUSkin [33]. In this paper, we focus on event-driven tactile learning with SNNs. Since the development of event-driven tactile sensors is still in its infancy [13], little prior work exists on learning event-based tactile data with SNNs. The work [1] employed a neural coding scheme to convert raw tactile data from non-event-based tactile sensors into event-based spike trains. It then utilized an SNN to process the spike trains and classify textures. A recent work [11] released the first publicly-available event-driven visual-tactile dataset collected by NeuTouch and proposed an SNN based on SLAYER [14] to solve the event-driven tactile learning. Moreover, to naturally capture the spatial topological relations and structural knowledge in the event-based tactile data, a very recent work [13] adopted the spiking graph neural network [15] to process the event-based tactile data and conduct the tactile object recognition. In this paper, different from previous works building SNNs with spiking neurons that employ the temporal recurrent neuronal dynamics, we construct SNNs with location spiking neurons to capture the complex spatio-temporal dependencies in the event-based tactile data and boost the event-driven tactile learning.

III. METHODS

In this section, we first demonstrate the spatial recurrent neuronal dynamics of location spiking neurons by introducing notations for the existing spiking neuron model – TSRM and extending it to the location spiking neuron model – LSRM. We then introduce models with location spiking neurons for

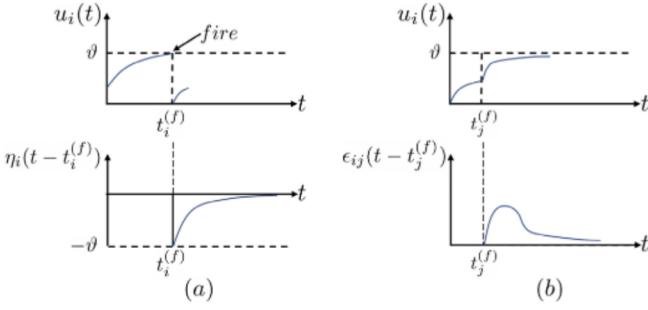


Fig. 1. The **temporal** recurrent neuronal dynamics of TSRM neuron i . (a) the refractory dynamics of TSRM neuron i . Immediately after firing an output spike at time $t_i^{(f)}$, the value of $u_i(t)$ is lowered or reset by adding a negative contribution $\eta_i(\cdot)$. The kernel $\eta_i(\cdot)$ vanishes for $t < t_i^{(f)}$ and decays to zero for $t \rightarrow \infty$. (b) the incoming spike dynamics of TSRM neuron i . A presynaptic spike at time $t_j^{(f)}$ increases the value of $u_i(t)$ for $t \geq t_j^{(f)}$ by an amount of $w_{ij}x_j(t_j^{(f)})\epsilon_{ij}(t - t_j^{(f)})$. The kernel $\epsilon_{ij}(\cdot)$ vanishes for $t < t_j^{(f)}$. A transmission delay may be included in the definition of $\epsilon_{ij}(\cdot)$.

event-driven tactile learning. Last, we provide implementation details and algorithms related to the proposed models.

A. Location Spiking Neurons

Spiking neuron models are mathematical descriptions of specific cells in the nervous system. They are the basic building blocks of SNNs. Two commonly-used spiking neuron models are the LIF model and the TSRM. Since the TSRM is more general than the LIF model [34], we introduce the TSRM and transform it to a location spiking neuron model – the LSRM. A similar transformation process can be applied to the LIF model to derive its corresponding location spiking neuron model.

In the TSRM, the temporal recurrent neuronal dynamics of neuron i are described by its membrane potential $u_i(t)$. When $u_i(t)$ exceeds a predefined threshold ϑ at the firing time $t_i^{(f)}$, the neuron i will generate a spike. The set of all firing times of neuron i is denoted by

$$\mathcal{F}_i = \{t_i^{(f)}; 1 \leq f \leq n\} = \{t | u_i(t) = \vartheta\}, \quad (1)$$

where $t_i^{(n)}$ is the most recent spike time $t_i^{(f)} < t$. The value of $u_i(t)$ is governed by two different spike response processes:

$$u_i(t) = \sum_{t_i^{(f)} \in \mathcal{F}_i} \eta_i(s_i) + \sum_{j \in \Gamma_i} \sum_{t_j^{(f)} \in \mathcal{F}_j} w_{ij} x_j(t_j^{(f)}) \epsilon_{ij}(s_j), \quad (2)$$

where $s_i = t - t_i^{(f)}$, $s_j = t - t_j^{(f)}$, Γ_i is the set of presynaptic neurons of neuron i , and $x_j(t_j^{(f)}) = 1$ is the presynaptic spike. $\eta_i(\cdot)$ is the refractory kernel, which describes the response of neuron i to its own spikes at time t . $\epsilon_{ij}(\cdot)$ is the incoming spike response kernel, which models the neuron i 's response to the presynaptic spikes from neuron j at time t . w_{ij} accounts for the connection strength between neuron i and neuron j and scale the incoming spike response. Figure 1(a) visualizes the refractory dynamics and Figure 1(b) visualizes the incoming spike dynamics. Without loss of generality, such temporal

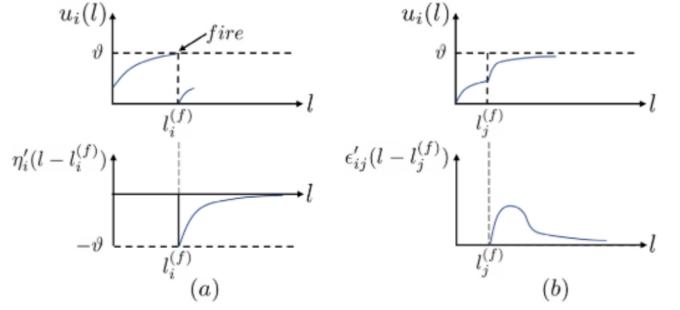


Fig. 2. The **spatial** recurrent neuronal dynamics of LSRM neuron i . (a) the refractory dynamics of LSRM neuron i . Immediately after firing an output spike at location $l_i^{(f)}$, the value of $u_i(l)$ is lowered or reset by adding a negative contribution $\eta'_i(\cdot)$. The kernel $\eta'_i(\cdot)$ vanishes for $l < l_i^{(f)}$ and decays to zero for $l \rightarrow \infty$. (b) the incoming spike dynamics of LSRM neuron i . A presynaptic spike at location $l_j^{(f)}$ increases the value of $u_i(l)$ for $l \geq l_j^{(f)}$ by an amount of $w'_{ij}x'_j(l_j^{(f)})\epsilon'_{ij}(l - l_j^{(f)})$. The kernel $\epsilon'_{ij}(\cdot)$ vanishes for $l < l_j^{(f)}$. A location delay may be included in the definition of $\epsilon'_{ij}(\cdot)$. “ $<$ ” and “ \geq ” indicate the location order.

recurrent neuronal dynamics also apply to other spiking neuron models, such as LIF neurons. From the above descriptions, we find that existing spiking neuron models **explicitly** convolve the temporal information in the data but fail to **explicitly** convolve the spatial information in the data, which, to some extent, limits their representative abilities.

To enrich the representative abilities of existing spiking neuron models, we propose location spiking neurons, which adopt the spatial recurrent neuronal dynamics and update their membrane potentials based on locations³. These neurons are able to **explicitly** convolve the spatial information in the data and enable us to extract features of event-based data in a novel way. Specifically, we adopt the TSRM and transform it to the LSRM. In the LSRM, the spatial recurrent neuronal dynamics of neuron i are described by its location membrane potential $u_i(l)$. When $u_i(l)$ exceeds a predefined threshold ϑ at the firing location $l_i^{(f)}$, the neuron i will generate a spike. The set of all firing locations of neuron i is denoted by

$$\mathcal{G}_i = \{l_i^{(f)}; 1 \leq f \leq n\} = \{l | u_i(l) = \vartheta\}, \quad (3)$$

where $l_i^{(n)}$ is the nearest firing location $l_i^{(f)} < l$. “ $<$ ” indicates the location order, which is manually set and will be discussed in Section III-B4. The value of $u_i(l)$ is governed by two different spike response processes:

$$u_i(l) = \sum_{l_i^{(f)} \in \mathcal{G}_i} \eta'_i(s'_i) + \sum_{j \in \Gamma'_i} \sum_{l_j^{(f)} \in \mathcal{G}_j} w'_{ij} x'_j(l_j^{(f)}) \epsilon'_{ij}(s'_j), \quad (4)$$

where $s'_i = l - l_i^{(f)}$, $s'_j = l - l_j^{(f)}$, Γ'_i is the set of presynaptic neurons of neuron i , and $x'_j(l_j^{(f)}) = 1$ is the presynaptic spike. $\eta'_i(\cdot)$ is the refractory kernel, which describes the response of neuron i to its own spikes at location l . $\epsilon'_{ij}(\cdot)$ is the incoming spike response kernel, which models the neuron i 's response

³locations could refer to pixel locations for images or taxel locations for tactile sensors.

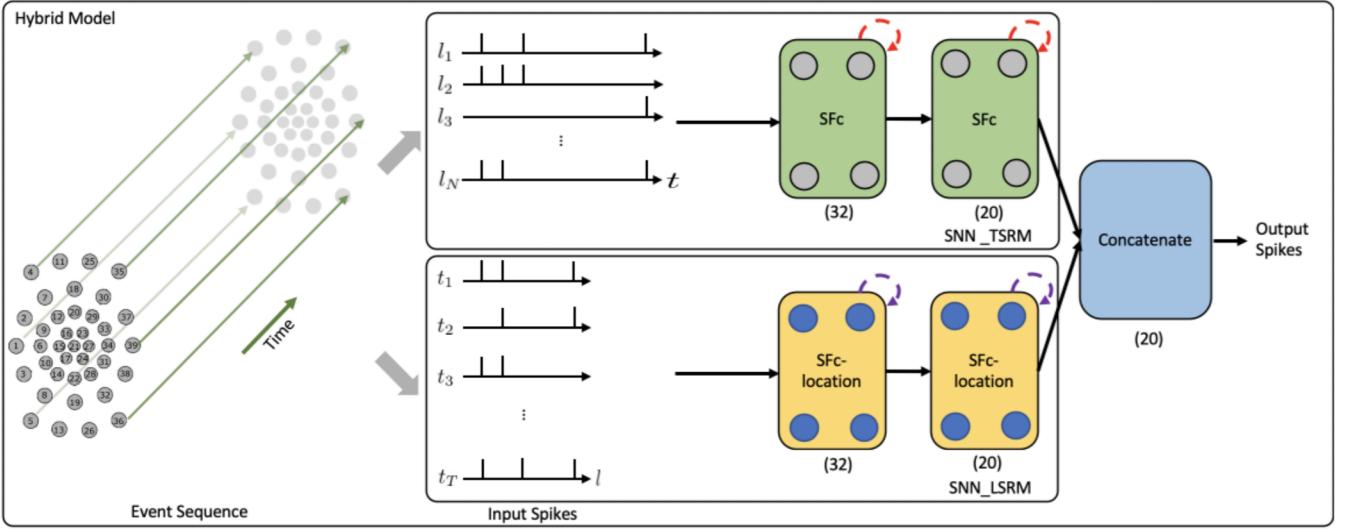


Fig. 3. The network structure of the hybrid model. The SNN with TSRM neurons (SNN_TSRM) processes the input spikes X_{in} and adopts the temporal recurrent neuronal dynamics (shown with red dashed arrows) of TSRM neurons to extract features from the data. The SNN with LSRM neurons (SNN_LSRM) processes the transposed input spikes X'_{in} and employs the spatial recurrent neuronal dynamics (shown with purple dashed arrows) of LSRM neurons to extract features from the data. Finally, the spiking representations from two networks are concatenated to yield the final predicted label. (32) and (20) represent the sizes of fully-connected layers, where we assume the number of classes (K) to be equal to 20.

to the presynaptic spikes from neuron j at location l . w_{ij} accounts for the connection strength between neuron i and neuron j and scale the incoming spike response. Figure 2(a) visualizes the refractory dynamics of LSRM neurons and Figure 2(b) visualizes the incoming spike dynamics of LSRM neurons. The threshold ϑ of LSRM neurons can be different from that of TSRM neurons, while we set the same for simplicity.

B. Event-Driven Tactile Learning with Location Spiking Neurons

Such location spiking neurons enable us to extract feature representations of event-based data in a novel way. To take advantage of location spiking neurons and boost the event-based tactile learning performance, we propose models with location spiking neurons, which capture complex spatio-temporal dependencies in the event-based tactile data. In this paper, we focus on processing the data collected by NeuTouch [11], a biologically-inspired event-driven fingertip tactile sensor with 39 taxels arranged spatially in a radial fashion (Fig. 3).

1) Hybrid Model: Figure 3 presents the network structure of the hybrid model. From the figure, we can see that the hybrid model has two components, including the SNN with TSRM neurons (SNN_TSRM) and the SNN with LSRM neurons (SNN_LSRM). Specifically, SNN_TSRM employs the temporal recurrent neuronal dynamics to extract spiking feature representations from the event-based tactile data $X_{in} \in \mathbb{R}^{N \times T}$, where N is the total number of taxels and T is the total time length of event sequences. SNN_LSRM utilizes the spatial recurrent neuronal dynamics to extract spiking feature representations from the event-based tactile data $X'_{in} \in \mathbb{R}^{T \times N}$, where X'_{in} is transposed from X_{in} . The spiking

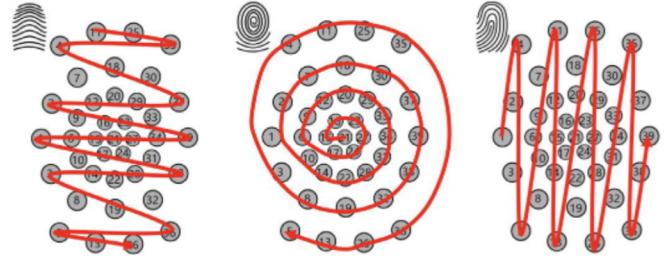


Fig. 4. Three bio-inspired location orders. Left to right: arch-like location order, whorl-like location order, loop-like location order.

representations from two networks are then concatenated to yield the final task-specific output.

2) SNN_TSRM vs. SNN_LSRM: The network structure of SNN_TSRM is shown in the top part of Fig. 3. It employs two spiking fully-connected layers with TSRM neurons (SFc) to process X_{in} and generate the spiking representations $O_1 \in \mathbb{R}^{K \times T}$, where K is the output dimension determined by the task. The membrane potential $u_i(t)$, the output spiking state $o_i(t)$, and the set of all firing times \mathcal{F}_i of TSRM neuron i in SFc are decided by:

$$u_i(t) = \sum_{t_j^{(f)} \in \mathcal{F}_i} \eta(s_i) + \underbrace{\sum_{j \in \Gamma_i} \sum_{t_j^{(f)} \in \mathcal{F}_j} w_{ij} o_j(t_j^{(f)}) \epsilon(s_j)}_{\text{capture spatial dependencies}},$$

$$o_i(t) = \begin{cases} 1 & \text{if } u_i(t) > \vartheta; \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

$$\mathcal{F}_i = \begin{cases} \mathcal{F}_i \cup t & \text{if } o_i(t) = 1; \\ \mathcal{F}_i & \text{otherwise,} \end{cases}$$

where w_{ij} are the trainable parameters, $\eta(\cdot)$ and $\epsilon(\cdot)$ are predefined by hyperparameters, Γ_i is the set of presynaptic neurons spanning over the spatial domain, which is utilized to capture the spatial dependencies in the event-based data.

The network structure of SNN_LSRM is shown in the bottom part of Fig. [3]. It employs two spiking fully-connected layers with LSRM neurons (SFC-location) to process X'_{in} and generate the spiking representations $O_2 \in \mathbb{R}^{K \times N}$, where K is the output dimension decided by the task. The membrane potential $u_i(l)$, the output spiking state $o_i(l)$, and the set of all firing locations \mathcal{G}_i of LSRM neuron i in SFC-location are decided by:

$$u_i(l) = \sum_{l_i^{(f)} \in \mathcal{G}_i} \eta'(s'_i) + \underbrace{\sum_{j \in \Gamma'_i} \sum_{l_j^{(f)} \in \mathcal{G}_j} w'_{ij} o_j(l_j^{(f)}) \epsilon'(s'_j)}_{\text{model temporal dependencies}},$$

$$o_i(l) = \begin{cases} 1 & \text{if } u_i(l) > \vartheta; \\ 0 & \text{otherwise,} \end{cases} \quad (6)$$

$$\mathcal{G}_i = \begin{cases} \mathcal{G}_i \cup l & \text{if } o_i(l) = 1; \\ \mathcal{G}_i & \text{otherwise,} \end{cases}$$

where w'_{ij} are the trainable parameters, $\eta'(\cdot)$ and $\epsilon'(\cdot)$ are predefined by hyperparameters, Γ'_i is the set of presynaptic neurons spanning over the temporal domain, which is utilized to model the temporal dependencies in the event-based data. Such location spiking neurons tap the representative potential and enable us to capture features in this way.

3) *Concatenate*: We concatenate the spiking representations of O_1 and O_2 along the last dimension and obtain the final output spike train $O \in \mathbb{R}^{K \times (T+N)}$. The predicted label is associated with the neuron $k \in K$ with the largest number of spikes in the duration of $T + N$.

4) *Location Orders*: To enable location spiking neurons' spatial recurrent neuronal dynamics, we propose three location orders for event-based tactile learning (Fig. [4]) based on three major fingerprint patterns of humans – arch, whorl, and loop. Three examples are shown here. Each number in the brackets represents the taxel index shown in Fig. [3]

- An example for the arch-like location order: [11, 25, 35, 4, 18, 30, 7, 2, 20, 37, 29, 12, 9, 33, 23, 16, 1, 6, 15, 21, 27, 34, 39, 24, 17, 10, 31, 38, 28, 14, 3, 22, 32, 8, 19, 36, 5, 13, 26]
- An example for the whorl-like location order: [21, 15, 16, 23, 27, 24, 17, 6, 9, 12, 20, 29, 33, 34, 31, 28, 22, 14, 10, 1, 2, 7, 18, 30, 37, 39, 38, 32, 19, 8, 3, 4, 11, 25, 35, 36, 26, 13, 5]
- An example for the loop-like location order: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39]

C. Implementation Details and Algorithms

Similar to the spike-count loss of prior works [11], [14], we propose a location spike-count loss to optimize the SNN with

LSRM neurons:

$$\mathcal{L}_{location} = \frac{1}{2} \sum_{k=0}^K \left(\sum_{l=0}^N o_k(l) - \sum_{l=0}^N \hat{o}_k(l) \right)^2, \quad (7)$$

which captures the difference between the observed output spike count $\sum_{l=0}^N o_k(l)$ and the desired spike count $\sum_{l=0}^N \hat{o}_k(l)$ across the K neurons. Moreover, to optimize the hybrid model, we develop a weighted spike-count loss:

$$\mathcal{L} = \frac{1}{2} \sum_{k=0}^K \left(\sum_{t=0}^T o_k(t) + \lambda \sum_{l=0}^N o_k(l) - \sum_{c=0}^{T+N} \hat{o}_k(c) \right)^2, \quad (8)$$

which first balances the contributions from two SNNs and then captures the difference between the observed balanced output spike count $\sum_{t=0}^T o_k(t) + \lambda \sum_{l=0}^N o_k(l)$ and the desired spike count $\sum_{c=0}^{T+N} \hat{o}_k(c)$ across the K output neurons. For both $\mathcal{L}_{location}$ and \mathcal{L} , the desired spike counts have to be specified for the correct and incorrect classes and are task-dependent hyperparameters. We set these hyperparameters like [11] for simplicity. To overcome the non-differentiability of spikes and apply the backpropagation algorithm, we use the approximate gradient proposed in SLAYER [14]. The timestep-wise inference algorithm of the hybrid model is shown in Alg. [1]. And the corresponding timestep-wise training algorithm can be derived by incorporating the weighted spike-count loss.

Algorithm 1 Timestep-wise inference algorithm

Input: event-based tactile inputs $X_{in} \in \mathbb{R}^{N \times T}$, N taxels, and the total time length T .

Output: timestep-wise predictions of O_1 , O_2 , and O .

```

1: for  $t \leftarrow 1$  to  $T$  do
2:   obtain  $X \in \mathbb{R}^{N \times t}$ 
3:   obtain  $\bar{X}' = \text{concatenate}(X', \mathbf{0}) \in \mathbb{R}^{T \times N}$ , where
    $X' \in \mathbb{R}^{t \times N}$ , and  $\mathbf{0} \in \mathbb{R}^{(T-t) \times N}$ 
4:    $O_1(t) = \mathbf{0} \in \mathbb{R}^{K \times t}$ ,  $O_2(t) = \mathbf{0} \in \mathbb{R}^{K \times N}$ 
5:    $O(t) = \mathbf{0} \in \mathbb{R}^{K \times (t+N)}$ 
6:    $O_1(t) = \text{SNN\_TSRM}(X)$ 
7:    $O_2(t) = \text{SNN\_LSRM}(\bar{X}')$ 
8:    $O(t) = \text{concatenate}(O_1(t), O_2(t))$ 
9: end for
```

IV. EXPERIMENTS

In this section, we first introduce the datasets and models for event-driven tactile learning. Next, to show the effectiveness of our proposed models, we extensively evaluate their performance on three event-driven tactile datasets and compare them with state-of-the-art models. Finally, we demonstrate the superior energy efficiency of our proposed models over ANNs and show the high-efficiency benefit of location spiking neurons. We utilize the slayerPytorch framework⁴ to implement the proposed models and employ RMSProp with the l_2 regularization to optimize them. The source code is available at <https://github.com/pkang2017/TactileLocNeurons>.

⁴<https://github.com/bamsumit/slayerPytorch>

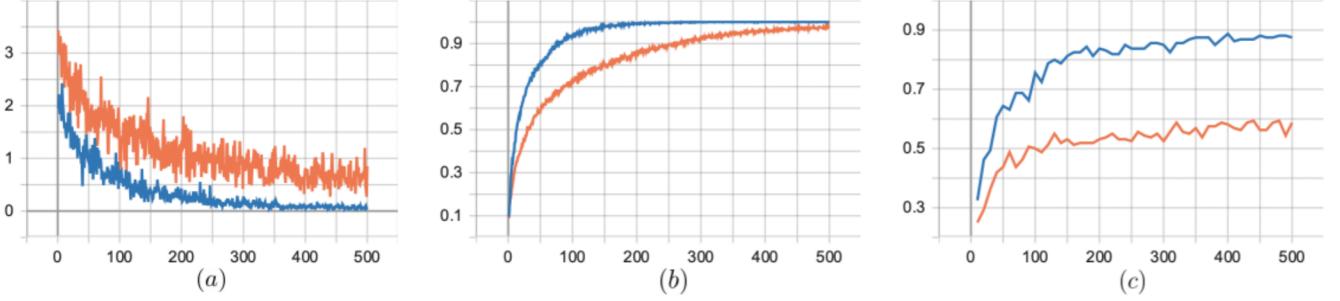


Fig. 5. Training and testing profiles for Tactile-SNN (orange) and Ours-Hybrid (blue): (a) the training loss, (b) the training accuracy, (c) the testing accuracy.

TABLE I
DATASET STATISTICS

Datasets	TD (s)	SSR (s)	T	N	K	#Samples
Objects	6.5	0.02	325	78	36	900
Containers	6.5	0.02	325	78	20	800
Slip Detection	0.15	0.001	150	78	2	100

TABLE II
ACCURACIES ON THE EVENT-DRIVEN TACTILE LEARNING DATASETS

Method	Type	Objects	Containers	Slip Detection
Tactile-SNN [11]	SNN	0.75	0.57*	0.82*
TactileSGNet [13]	SNN	0.79	0.58	0.97
GRU-MLP [11]	ANN	0.72	0.46*	0.87*
CNN-3D [11]	ANN	0.90	0.67*	0.44*
Ours-Hybrid	SNN	0.91	0.86	1.0

*These values come from the paper [11]. And the best performance is in bold.

A. Datasets

In this paper, we use the datasets collected by NeuTouch [11]. Specifically, three datasets are collected, including “Objects” and “Containers” for event-driven tactile object recognition and “Slip Detection” for event-driven slip detection. Unlike “Objects” only requiring models to determine the type of objects being handled, “Containers” asks models about the type of containers being handled and the amount of liquid (0%, 25%, 50%, 75%, 100%) held within. Thus, “Containers” is more challenging for event-driven tactile object recognition. Moreover, the task of event-driven slip detection is also challenging since it requires models to detect the rotational slip within a short time, like 0.15s for “Slip Detection”. We summarize the dataset statistics in Table I where TD is for time durations, SSR is for spike sampling rates, $T=TD / SSR$ is the total time length, N is the total number of taxels (each tactile sensor has 39 taxels and two tactile sensors are used), and K is the number of classes. We split the data into a training set (80%) and a test set (20%) with an equal class distribution in the experiments. We repeat each experiment for five rounds and report the average accuracy.

B. Models

We compare our models with the state-of-the-art SNN methods for event-driven tactile learning, including Tactile-

TABLE III
ACCURACIES FOR ABLATION STUDIES

Method	Type	Objects	Containers	Slip Detection
Tactile-SNN [11]	SNN	0.75	0.57	0.82
Ours-Location Tactile-SNN	SNN	0.89	0.88	0.82
Ours-Hybrid $\lambda = 1$	SNN	0.91	0.86	1.0
Ours-Hybrid $\lambda = 0.5$	SNN	0.92	0.89	0.98
Ours-Hybrid-loop	SNN	0.91	0.86	1.0
Ours-Hybrid-arch	SNN	0.91	0.86	0.99
Ours-Hybrid-whorl	SNN	0.92	0.86	0.98

SNN [11] and TactileSGNet [13]. Tactile-SNN employs **TSRM neurons** as the building blocks, and the network structure of Tactile-SNN is Input-SFc0-SFc1. While TactileSGNet utilizes **LIF neurons** as the building blocks and proposes the spiking graph neural network (SGNet). The network structure of TactileSGNet is Input-SGNet-SFc1-SFc2-SFc3. We also compare our models against conventional deep learning, specifically Gated Recurrent Units (GRUs) [35] with Multi-layer Perceptrons (MLPs) and 3D convolutional neural networks [36]. The network structure of GRU-MLP is Input-GRU-MLP, where MLP is only utilized at the final time step. And the network structure of CNN-3D is Input-3D_CNN1-3D_CNN2-Fc1.

C. Performance and Analysis

1) *Basic Performance:* Table II presents the test accuracies on the three datasets. We observe that our hybrid model significantly outperforms the state-of-the-art SNNs. Moreover, figure 5 shows the training and testing profiles for Tactile-SNN and our hybrid model. From this figure, we can see that our model converges faster and attains the lower loss and the higher accuracy compared to Tactile-SNN. The reason why our model is superior to other SNNs could be two-fold: (1) different from state-of-the-art SNNs that only extract features with existing spiking neurons, our model employs an SNN with location spiking neurons to extract features in a novel way; (2) our model fuses SNN_TSRM and SNN_LSRM to better capture complex spatio-temporal dependencies in the data. We also compare our model with ANNs, which provide fair comparison baselines for fully ANN architectures since they employ similar lightsome network architectures as ours. From Table II we find out that our model outperforms ANNs

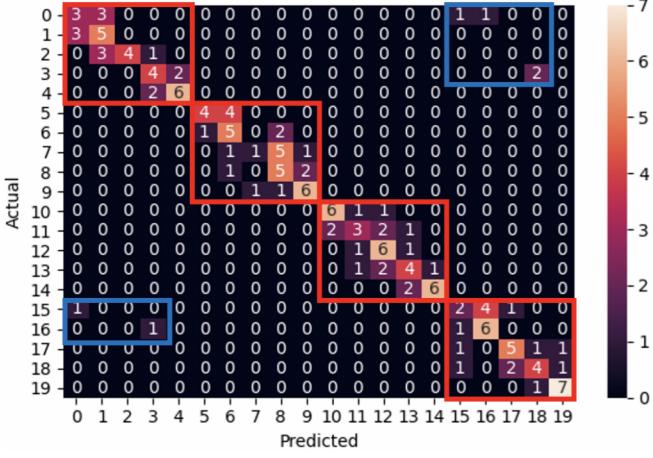


Fig. 6. The confusion matrix of Tactile-SNN on “Containers”.

on the three tasks, which might be because our model is more compatible with various kinds of event-based data and better maintains the sparsity to prevent overfitting.

2) *Ablation Studies*: To examine the effectiveness of each component in the hybrid model, we **separately train** SNN_TSRM (which is exactly Tactile-SNN) and SNN_LSRM (which is referred to as Location Tactile-SNN). From Table III, we surprisingly find out that Location Tactile-SNN significantly surpasses Tactile-SNN on the datasets for event-driven tactile object recognition and provides comparable performance on the event-driven slip detection. The reason for this could be two-fold: (1) the time durations of event-driven tactile object recognition datasets are longer than that of “Slip Detection”, and Location Tactile-SNN is good at capturing the mid-and-long term dependencies in these object recognition datasets; (2) like Tactile-SNN, Location Tactile-SNN can still capture the spatial dependencies in the event-driven tactile data (“Slip Detection”) due to the spatial recurrent neuronal dynamics of location spiking neurons. Furthermore, we examine the sensitivities of λ in Eq.(8) and location orders. From Table III, we notice the results of related models are close, proving that the λ tuning and location orders do not significantly impact the task performance.

3) *Confusion Matrices*: We calculate the confusion matrices of Tactile-SNN (Fig. 6) and our hybrid model (Fig. 7) on “Containers” since it is a more challenging event-driven tactile object recognition dataset. From the two figures, we can see that our hybrid model can perfectly distinguish the different containers. Each red box in the figures represents a type of container, and each blue box in the figures represents the container misclassification. Moreover, compared to Tactile-SNN, we observe that our model can recognize the container fullness with a higher accuracy since the misclassification number in each red box is fewer for our model.

4) *Timestep-wise Inference*: We evaluate the timestep-wise inference performance of the hybrid model and validate the contributions of the two components in it. Moreover, we

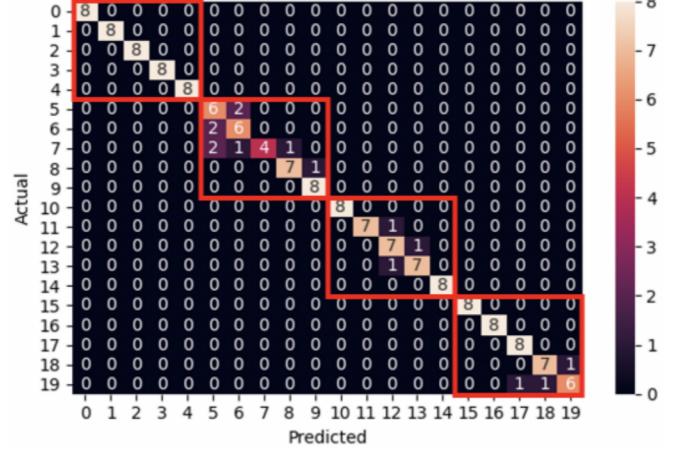


Fig. 7. The confusion matrix of Ours-Hybrid on “Containers”.

propose a time-weighted hybrid model to better balance the two components’ contributions and achieve the better overall performance. Figure 8(a), 8(b), and 8(c) show the timestep-wise inference accuracies (%) of SNN_TSRM, SNN_LSRM, the hybrid model, and the time-weighted hybrid model on the three datasets. Specifically, the output of the time-weighted hybrid model at time t is

$$O_{tw}(t) = \text{concatenate}((1 - \omega) * O1(t), \omega * O2(t)), \quad (9)$$

$$\omega = \frac{1}{1 + e^{-\psi * (\frac{t}{T} - 1)}},$$

where the hyperparameter ψ controls the balance between SNN_TSRM’s contribution and SNN_LSRM’s contribution and T is the total time length. From the figures, we can see that SNN_TSRM has good “early” accuracies on the three tasks since it well captures the spatial dependencies with the help of Eq. (5). However, its accuracies do not improve too much at the later stage since it does not sufficiently capture the temporal dependencies. In contrast, SNN_LSRM has fair “early” accuracies, while its accuracies jump a lot at the later stage since it models the temporal dependencies in Eq. (6). The hybrid model adopts the advantages of these two components and extracts spatio-temporal features from various views, which enables it to have a better overall performance. Furthermore, after employing the time-weighted output and shifting more weights to SNN_TSRM at the early stage, the time-weighted hybrid model can have a good “early” accuracy as well as an excellent “final” accuracy.

D. Energy Efficiency

Following the estimation method in [15], [37]⁵ we estimate the computational costs of the hybrid model and ANNs on the three datasets. As shown in Fig. 8(d), the hybrid model has no multiplication operations and achieves far fewer addition operations than ANN models on the three datasets. Moreover, based on Table IV, the compression ratio of total operations

⁵We consider the computational costs in feature matrix transformation.

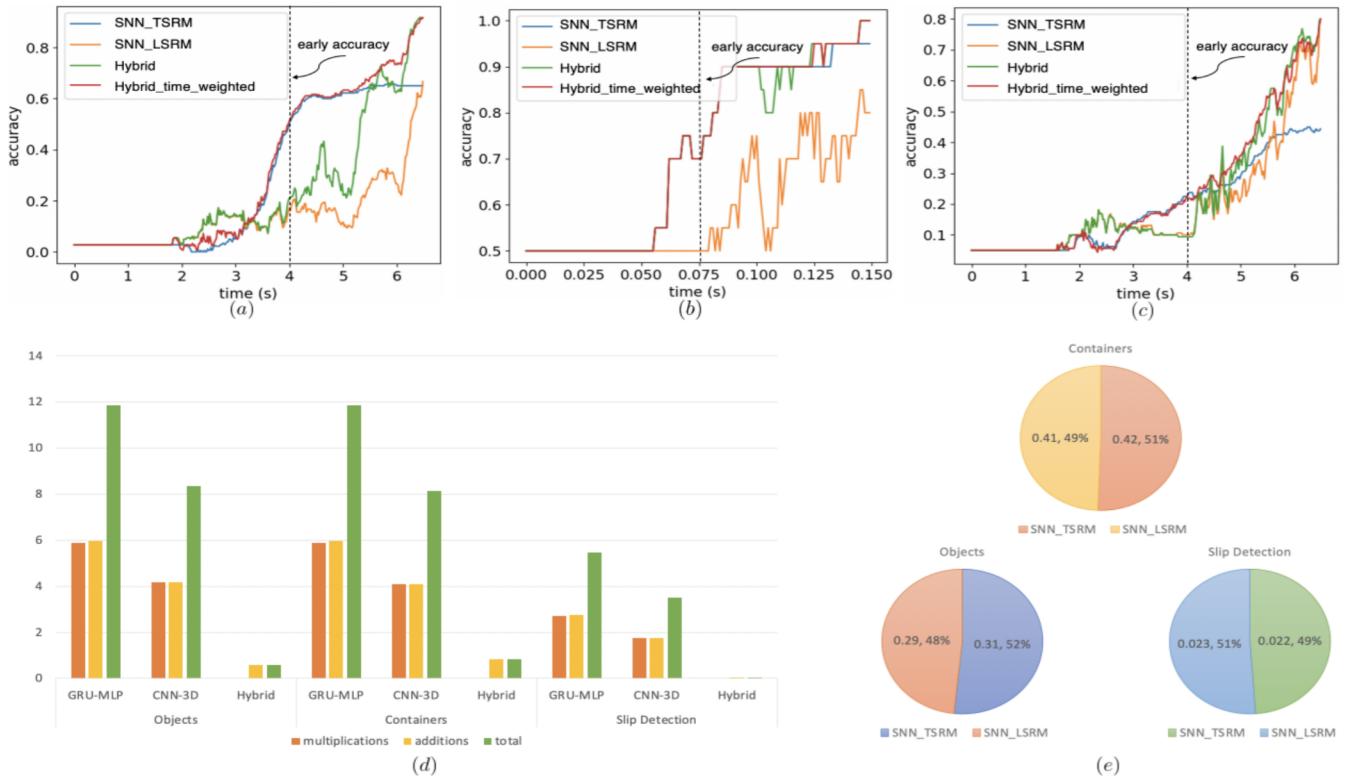


Fig. 8. The timestep-wise inference (Alg. 1 for SNN_TSRM – O_1 , SNN_LSRM – O_2 , the hybrid model – O , and the time-weighted hybrid model – O_{tw}) on (a) “Objects”, (b) “Slip Detection”, (c) “Containers”. Note that we use the same event sequences as [11] and the first spike occurs at around 2.0s for “Objects” and “Containers”. (d) Operation cost ($\times 10^6$) comparison between ANNs and Ours-Hybrid on the benchmark datasets. (e) Operation cost ($\times 10^6$) and percentage comparison between SNN_TSRM and SNN_LSRM on the benchmark datasets.

TABLE IV
OPERATION COST ($\times 10^6$) AND COMPRESSION RATIO ON BENCHMARK DATASETS

Method	Objects	Containers	Slip Detection
GRU-MLP	11.87	11.87	5.48
CNN-3D	8.34	8.14	3.5
Ours-Hybrid	0.60	0.83	0.045
Ratio	9.80~14.30 \times	13.90~19.78 \times	77.78~121.78 \times

(ANNs Opts. / Ours Opts.) is between **9.80 \times** and **121.78 \times** . These results are consistent with the fact that the sparse spike communication and event-driven computation underlie the efficiency advantage of SNNs and demonstrate the potentials of our model on neuromorphic hardware. We further compare the costs of SNN_TSRM and SNN_LSRM on the benchmark datasets. From Fig. [8(e)], we can see that the cost of SNN_LSRM is almost equal to that of SNN_TSRM on each dataset, which shows that the location spiking neurons have the similar energy efficiency compared to existing spiking neurons. Such high-efficiency benefits make location spiking neurons a perfect fit for neuromorphic hardware.

V. DISCUSSION AND CONCLUSION

This paper proposes a novel neuron model – “location spiking neuron” and introduces the spatial recurrent neuronal

dynamics of LSRM neurons. We believe the idea of such location spiking neurons can be applied to other existing spiking neuron models like LIF neurons and strengthen their feature representation abilities. Moreover, we think the location spiking neurons can build more complicated models to further boost the event-driven tactile learning performance. For example, we can develop a spiking graph neural network with location spiking neurons and combine it with [13] to better serve event-driven tactile learning tasks. Furthermore, besides event-driven tactile learning, we can apply the models with location spiking neurons to other event-driven learning fields, like event-based vision or event-driven audio sensing. By analyzing the applications in these fields, we can further understand the strengths and weaknesses of this new neuron.

In this work, we propose location spiking neurons and demonstrate the dynamics of LSRM neurons. By exploiting the LSRM neurons, we develop several models for event-driven tactile learning to sufficiently capture the complex spatio-temporal dependencies. The experimental results on three datasets demonstrate the extraordinary performance and high energy efficiency of our models and location spiking neurons. This further unlocks their potential on neuromorphic hardware. Overall, this work sheds new light on SNN representation learning and event-driven learning, which may facilitate the understanding of advanced cognitive intelligence.

REFERENCES

- [1] Tasbolat Taunyazov, Yansong Chua, Ruihan Gao, Harold Soh, and Yan Wu, "Fast texture classification using tactile neural coding and spiking neural network," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9890–9895.
- [2] Harold Soh and Yiannis Demiris, "Incrementally learning objects by touch: Online discriminative and generative models for tactile-based recognition," *IEEE transactions on haptics*, vol. 7, no. 4, pp. 512–525, 2014.
- [3] Zhanat Kappassov, Juan-Antonio Corrales, and Véronique Perdereau, "Tactile sensing in dexterous robot hands," *Robotics and Autonomous Systems*, vol. 74, pp. 195–220, 2015.
- [4] Jose Sanchez, Carlos M Mateo, Juan Antonio Corrales, Belhassen-Chedli Bouzgarrou, and Youcef Mezouar, "Online shape estimation based on tactile sensing and deformation modeling for robot manipulation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 504–511.
- [5] Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H Adelson, and Sergey Levine, "More than a feeling: Learning to grasp and regrasp using vision and touch," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3300–3307, 2018.
- [6] Shiv S Baishya and Berthold Bäuml, "Robust material classification with a tactile skin using deep learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 8–15.
- [7] Tasbolat Taunyazov, Hui Fang Koh, Yan Wu, Caixia Cai, and Harold Soh, "Towards effective tactile identification of textures using a hybrid touch approach," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4269–4275.
- [8] Da Li, Xinbo Chen, Michela Beccati, and Ziliang Zong, "Evaluating the energy efficiency of deep convolutional neural networks on cpus and gpus," in *2016 IEEE international conferences on big data and cloud computing (BDCloud), social computing and networking (SocialCom), sustainable computing and communications (SustainCom)(BDCloud-SocialCom-SustainCom)*. IEEE, 2016, pp. 477–484.
- [9] Emma Strubell, Ananya Ganesh, and Andrew McCallum, "Energy and policy considerations for deep learning in nlp," *arXiv preprint arXiv:1906.02243*, 2019.
- [10] Guillermo Gallego, Tobi Delbrück, Garrick Michael Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew Davison, Jorg Conradt, Kostas Daniilidis, et al., "Event-based vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, 2020.
- [11] Tasbolat Taunyazov, Weicong Sng, Hian Hian See, Brian Lim, Jethro Kuan, Abdul Fatir Ansari, Benjamin Tee, and Harold Soh, "Event-driven visual-tactile sensing and learning for robots," in *Proceedings of Robotics: Science and Systems*, July 2020.
- [12] Michael Pfeiffer and Thomas Pfeil, "Deep learning with spiking neurons: opportunities and challenges," *Frontiers in neuroscience*, vol. 12, pp. 774, 2018.
- [13] Fuqiang Gu, Weicong Sng, Tasbolat Taunyazov, and Harold Soh, "Tactilesnet: A spiking graph neural network for event-based tactile object recognition," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9876–9882.
- [14] Sumit Bam Shrestha and Garrick Orchard, "SLAYER: Spike layer error reassignment in time," in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., pp. 1419–1428. Curran Associates, Inc., 2018.
- [15] Mingkun Xu, Yujie Wu, Lei Deng, Faqiang Liu, Guoqi Li, and Jing Pei, "Exploiting spiking dynamics with spatial-temporal feature normalization in graph learning," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, Zhi-Hua Zhou, Ed. 8 2021, pp. 3207–3213, International Joint Conferences on Artificial Intelligence Organization, Main Track.
- [16] Andrew L Maas, Awini Y Hannun, Andrew Y Ng, et al., "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*. Citeseer, 2013, vol. 30, p. 3.
- [17] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.
- [18] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.
- [19] Wulfraum Gerstner, "Time structure of the activity in neural network models," *Physical review E*, vol. 51, no. 1, pp. 738, 1995.
- [20] Larry F Abbott, "Lapicque's introduction of the integrate-and-fire model neuron (1907)," *Brain research bulletin*, vol. 50, no. 5-6, pp. 303–304, 1999.
- [21] Wulfraum Gerstner and Werner M Kistler, *Spiking neuron models: Single neurons, populations, plasticity*, Cambridge university press, 2002.
- [22] Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda, "Towards spike-based machine intelligence with neuromorphic computing," *Nature*, vol. 575, no. 7784, pp. 607–617, 2019.
- [23] Ed Bullmore and Olaf Sporns, "The economy of brain network organization," *Nature Reviews Neuroscience*, vol. 13, no. 5, pp. 336–349, 2012.
- [24] Daniel J Felleman and David C Van Essen, "Distributed hierarchical processing in the primate cerebral cortex," in *Cereb cortex*. Citeseer, 1991.
- [25] Paul A Merolla, John V Arthur, Rodrigo Alvarez-Icaza, Andrew S Cassidy, Jun Sawada, Filipp Akopyan, Bryan L Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, et al., "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014.
- [26] Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaia, Gabriel A Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R Risbud, "Advancing neuromorphic computing with loihi: A survey of results and outlook," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 911–934, 2021.
- [27] Yongqiang Cao, Yang Chen, and Deepak Khosla, "Spiking deep convolutional neural networks for energy-efficient object recognition," *International Journal of Computer Vision*, vol. 113, no. 1, pp. 54–66, 2015.
- [28] Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy, "Going deeper in spiking neural networks: Vgg and residual architectures," *Frontiers in neuroscience*, vol. 13, pp. 95, 2019.
- [29] Yuje Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi, "Spatiotemporal backpropagation for training high-performance spiking neural networks," *Frontiers in neuroscience*, vol. 12, pp. 331, 2018.
- [30] Xiang Cheng, Yunzhe Hao, Jiaming Xu, and Bo Xu, "Lisnn: Improving spiking neural networks with lateral interactions for robust object recognition," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, Christian Bessiere, Ed. 7 2020, pp. 1519–1525, International Joint Conferences on Artificial Intelligence Organization, Main track.
- [31] Alexander Schmitz, Marco Maggiali, Lorenzo Natale, Bruno Bonino, and Giorgio Metta, "A tactile sensor for the fingertips of the humanoid robot icub," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 2212–2217.
- [32] Jeremy A Fishel and Gerald E Loeb, "Sensing tactile microvibrations with the biotac—comparison with human sensitivity," in *2012 4th IEEE RAS & EMBS international conference on biomedical robotics and biomechatronics (BioRob)*. IEEE, 2012, pp. 1122–1127.
- [33] Tasbolat Taunyazov, Luar Shui Song, Eugene Lim, Hian Hian See, David Lee, Benjamin CK Tee, and Harold Soh, "Extended tactile perception: Vibration sensing through tools and grasped objects," *arXiv preprint arXiv:2106.00489*, 2021.
- [34] Wolfgang Maass and Christopher M Bishop, *Pulsed neural networks*, MIT press, 2001.
- [35] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
- [36] Juan M Gandarias, Francisco Pastor, Alfonso J García-Cerezo, and Jesús M Gómez-de Gabriel, "Active tactile recognition of deformable objects with 3d convolutional neural networks," in *2019 IEEE World Haptics Conference (WHC)*. IEEE, 2019, pp. 551–555.
- [37] Chanyu Lee, Adarsh Kumar Kosta, Alex Zihao Zhu, Kenneth Chaney, Kostas Daniilidis, and Kaushik Roy, "Spike-flownet: event-based optical flow estimation with energy-efficient hybrid neural networks," in *European Conference on Computer Vision*. Springer, 2020, pp. 366–382.