An Efficient Threshold-Driven Aggregate-Label Learning Algorithm for Multimodal Information Processing

Malu Zhang, Xiaoling Luo, Jibin Wu, Yi Chen, Ammar Belatreche, *Member, IEEE*, Zihan Pan, Hong Qu, *Member, IEEE* and Haizhou Li, *Fellow, IEEE*

***Abstract*—The aggregate-label learning paradigm tackles the long-standing temporary credit assignment (TCA) problem in neuroscience and machine learning, enabling spiking neural networks to learn multimodal sensory clues with delayed feedback signals. However, the existing aggregate-label learning algorithms only work for single spiking neurons, and with low learning efficiency, which limit their real-world applicability. To address these limitations, we first propose an efficient threshold-driven plasticity algorithm for spiking neurons, namely ETDP. It enables spiking neurons to generate the desired number of spikes that match the magnitude of delayed feedback signals and to learn useful multimodal sensory clues embedded within spontaneous spiking activities. Furthermore, we extend the ETDP algorithm to support multi-layer spiking neural networks (SNNs), which significantly improves the applicability of aggregate-label learning algorithms. We also validate the multi-layer ETDP learning algorithm in a multimodal computation framework for audio-visual pattern recognition. Experimental results on both synthetic and realistic datasets show significant improvements in the learning efficiency and model capacity over the existing aggregate-label learning algorithms. It, therefore, provides many opportunities for solving real-world multimodal pattern recognition tasks with spiking neural networks.**

***Index Terms*—Spiking neurons, spiking neural networks, aggregate-label learning, synaptic plasticity, multimodal information**

I. Introduction

THE brain has a remarkable ability to integrate multimodal sensory information for efficient detection and identification of different external events, so as to adaptively interact with the environment. For example, when a predator approaches its prey, the sounds of breaking twigs and the odor of the predator represent essential survival clues for the prey [1]. Life becomes much easier when an individual learns these multi-sensory clues. However, it remains challenging for biological neural systems to learn these useful multi-sensory clues because they are usually embedded within distracting streams of unrelated sensory signals. Even worse, the feedback signals typically arrive after long and variable delays [1][2]. Learning useful multi-sensory clues requires bridging the gap between their occurrence and the delayed arrival of feedback signals [1][2][3]. This challenge, known as the temporal credit-assignment (TCA) problem, is one of the long-standing research topics in neuroscience and machine learning. While it remains unclear how the brain resolves this challenging TCA problem, the critical role of neural spikes (action potentials) in transmitting information and modulating learning in the brain is well recognized [4][5][6][7]. In recent years, many spike-based supervised learning algorithms have been proposed to explore the mechanisms underlying brain plasticity. Existing supervised learning methods aim to train output neurons to produce the desired spiking activity in response to an input spike pattern and are classified, depending on the number of target output spikes, into single- or multi-spike learning algorithms.

Tempotron [6] is one of the most popular single-spike learning algorithms, whereby synaptic weights are modified to ensure the learning neuron fires at least one spike when the desired input pattern is present and remains silent otherwise. Rank-order learning [8][9][10] is another type of single-spike learning algorithms, which adjusts synaptic weights to make the learning neuron fires the earliest spike in response to the desired input spike pattern. Subsequently, the time-to-first spike decoding strategy is employed in the output layer for rapid decision-making. The SpikeProp [11] learning algorithm constructs an error function from the distance between the times of the desired and the actual output spike and applies a modified error back-propagation (BP) algorithm to update synaptic weights. Although single-spike learning algorithms were successfully applied in many application domains [12][13][14], the spiking neural networks (SNNs) trained by these algorithms have limited storage capacity and are sensitive to noise.

In order to overcome these limitations, many multi-spike learning algorithms have been proposed in recent years. One

This research is supported by Programmatic Grant No. A1687b0033 from the Singapore Government's Research, Innovation and Enterprise 2020 plan (Advanced Manufacturing and Engineering domain), the National Natural Science Foundation of China (Grant No. 61976043 and 61573081), the Zhejiang Lab (Grant No. 2019KC0AB02), and the Zhejiang Lab's International Talent Fund for Young Professionals. *Corresponding author: Jibin Wu*, email: jibin.wu@u.nus.edu

M. Zhang is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China, and with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore (e-mail: maluzhang@nus.edu.sg)

J. Wu, Z. Pan, and H. Li are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore (e-mail: haizhou.li@nus.edu.sg)

X. Luo, Y. Chen, and H. Qu are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China.

A. Belatreche is with the Department of Computer and Information Sciences, Faculty of Engineering and Environment, Northumbria University, Newcastle upon Tyne NE1 8ST, U.K.

Malu Zhang, Xiaoling Luo and Yi Chen contributed equally in this work, and should be regarded as co-first authors.

well-known multi-spike learning algorithm is the remote supervised method (ReSuMe) [15]. In ReSuMe, the synaptic changes are driven by a combination of spike time-dependent plasticity (STDP) and anti-STDP. DL-ReSuMe [16] improves the learning performance of ReSuMe by considering both the synaptic plasticity and the delay plasticity.The chronotron [17] and the Spike Pattern Association Neuron (SPAN) [18] update the synaptic weights based on the distance defined by the Victor and Purpura metric [19] and the van Rossum metric [20], respectively. Besides these spike-driven learning algorithms, the membrane potential-driven methods are also proposed [7][21][22][23][24]. They utilize the neuron membrane potential to guild the target neurons, such that they fire at the desired times. Experimental results [7][21] suggest that the membrane potential-driven learning algorithms are more efficient than the spike-driven learning algorithms. The aforementioned multi-spike learning algorithms are only applicable when the desired spike times are provided. However, such information is often unavailable in neural systems and real application scenarios.

To circumvent this limitation, Gütig [1] puts forward a novel aggregate-label learning paradigm for spiking neurons, which trains spiking neurons to fire a desired number of spikes without considering the precise timing of each spike. Following this paradigm, several learning algorithms have been proposed and can be categorized into threshold-driven and membrane potential-driven. Multi-spike Tempotron (MST) [1], the first introduced threshold-driven method, transforms the discrete-valued spike count distance into a continuous-valued distance between the fixed biological firing threshold and the hypothetical threshold. With this transformation, the gradient descent method can be applied to optimize the synaptic weights by minimizing the firing threshold distance. As demonstrated in [1], the spiking neurons trained with the MST method can produce a desired number of spikes and learn predictive clues embedded within a long stream of unrelated spiking activities. Yu et al. [3] [25] propose another threshold-driven plasticity algorithm, namely TDP, which simplifies the recursive gradient computation of MST. Although the experimental results have shown improved learning efficiency over the MST, the approximated gradients derived by the TDP diverge from the theoretical ones with an increasing number of desired spike count, hence deteriorates the learning effectiveness.

On the other hand, the membrane potential-driven aggregate-label learning algorithms construct an error function between the membrane potential and the fixed biological firing threshold. Examples of this class of learning algorithms include MPD-AL [2] and its variants [26][27]. The membrane potential-driven algorithms have shown superior learning efficiency over their threshold-driven counterparts. However, the learning mechanism of these learning methods fails when a sub-threshold membrane potential peak is absent in between any two adjacent output spikes. In addition, membrane potential-driven algorithms impose some restrictions on the training samples when learning predictive clues [2], and are only applicable to single neurons.

This work attempts to improve the learning effectiveness and efficiency of the existing aggregate-label learning algorithms. We first propose an Efficient Threshold-Driven Plasticity (ETDP) algorithm for spiking neurons, which enables spiking neurons to generate the desired number of spikes that match the magnitude of delayed feedback. Furthermore, the proposed learning algorithm is capable of learning useful multi-sensory clues embedded within a long stream of distracting sensory activities. Besides, we introduce an exploding gradient prevention strategy (EGPS) to address the exploding gradient problem found in existing aggregate-label learning algorithms. Experimental results demonstrate that the ETDP learning algorithm significantly outperforms its counterparts in terms of learning efficiency. We further extend the ETDP algorithm to support multi-layer spiking neural networks, which significantly improves the computational capacity of the trained SNN models. We also validate the multi-layer ETDP learning algorithm in a multimodal computation framework for audio-visual pattern recognition. Experimental results on the MNIST and TIDIGITS datasets show that the proposed SNN-based multimodal recognition framework can improve the classification accuracy compared to its unimodal parts.

II. Neuron Model and ETDP Learning Algorithms

In this section, we first introduce the spiking neuron model adopted in this work. Then, we present the proposed ETDP algorithm for single spiking neurons and compare it to other existing aggregate-label learning algorithms. Finally, we extend the proposed ETDP algorithm for multi-layer spiking neural networks.

A. Neuron Model

In this work, we employ the current-based leaky integrate-and-fire (LIF) model to derive the proposed learning algorithm [1] due to its biological plausibility and computational tractability. We consider an output spiking neuron, connected with N afferent neurons, whose membrane potential is denoted by $V(t)$ and the resting potential V_{rest} is set to 0. Each incoming spike from the afferent neurons induces a postsynaptic potential (PSP) and integrated by the output neuron. The output neuron fires a spike when $V(t)$ reaches the firing threshold ϑ from below. The membrane potential dynamics of the LIF neuron can be expressed as

$$V(t) = V_{rest} + \sum_{i}^{N} \omega_i \sum_{t_i^j < t} K(t - t_i^j) - \vartheta \sum_{t_s^j < t} \exp\big(-\frac{t - t_s^j}{\tau_m}\big) \tag{1}$$

where ω_i is the synaptic weight of afferent i, and t_i^j denotes the j^{th} spike time of afferent i. The PSP kernel $K(t - t_i^j)$ is defined as

$$K(t - t_i^j) = V_0 \left[\exp\left(-\frac{t - t_i^j}{\tau_m}\right) - \exp\left(-\frac{t - t_i^j}{\tau_s}\right) \right] \tag{2}$$

where the integration time constant of the postsynaptic membrane τ_m and the decay time constant of synaptic currents τ_s jointly govern the shape of the PSP kernel. V_0 is a normalization constant that ensures a unitary peak value for

the PSP kernel. The last term in Eq. 1 is the refractory kernel, which resets the membrane to its resting potential after the spike generation. t_s^j denotes the time of the jth output spike.

B. ETDP Learning Algorithm for Single Spiking Neurons

The goal of the proposed ETDP learning algorithm is to modify the synaptic weights so that the trained neuron can fire the desired spike count. Due to the discrete nature of the spike count, its derivative with respect to synaptic weights cannot be obtained directly. To circumvent this problem, we apply the spike-threshold-surface (STS) to map the discrete spike counts to continuous hypothetical firing thresholds [1]. As shown in Fig. 1(b), the critical threshold ϑ_k^* denotes the threshold value at which the spike count jumps from $k-1$ to k. For example, given a particular input spike pattern and a set of synaptic weights, Fig. 1(a) shows that the neuron fires three spikes with the neuron's biological firing threshold $\vartheta = 1$ (red line). While the neuron fires four spikes (blue line) when the threshold decreases to ϑ_4^*. Based on the relationship between the STS and the number of output spikes, the problem of training a neuron to output the desired spike count d could be transformed into adjusting the STS so that $\vartheta_{d+1}^* < \vartheta \leq \vartheta_d^*$. Hence, the goal is violated either $\vartheta_{d+1}^* \geq \vartheta$ or $\vartheta_d^* < \vartheta$.

In general, there are two strategies to optimize the STS. One is the "absolute" rule that directly uses ϑ_{d+1}^* and ϑ_d^* to calculate the synaptic updates; while the other one uses the actual output spike count o to determine the synaptic updates, namely the "relative" rule. The "absolute" and the "relative" rules are summarized in Eq. 3 and Eq. 4, respectively.

$$\Delta\omega = \begin{cases} -\lambda \frac{d\vartheta_{d+1}^*}{d\omega} & \text{if} \quad \vartheta_{d+1}^* \geq \vartheta \\ \lambda \frac{d\vartheta_d^*}{d\omega} & \text{if} \quad \vartheta_d^* < \vartheta \end{cases} \tag{3}$$

$$\Delta\omega = \begin{cases} -\lambda \frac{d\vartheta_o^*}{d\omega} & \text{if} \quad o > d \\ \lambda \frac{d\vartheta_{o+1}^*}{d\omega} & \text{if} \quad d > o \end{cases} \tag{4}$$

where λ is the learning rate. It is worth noting that the absolute learning rule requires the exact value of desired spike counts, while the relative learning rule is based on a binary feedback signal that only specifies whether the neuron should increase or decrease the spike count. Therefore, the relative learning rule is simpler and biologically more plausible [3]. Therefore, we derive the proposed ETDP learning algorithm based on this relative learning rule.

According to the definition of critical threshold ϑ^*, there exists a unique t^* that satisfies

$$\vartheta^* = V(t^*) = V_o(t^*) - \vartheta^* \sum_{j=1}^{m} \exp\left(-\frac{t^* - t_s^j}{\tau_m}\right) \tag{5}$$

with

$$V_o(t^*) = \sum_i^N \omega_i \sum_{t_i^j < t^*} K(t^* - t_i^j) \tag{6}$$

Here, m denotes the total number of output spikes fired before t^*. Since ϑ^* depends on the synaptic weights also through

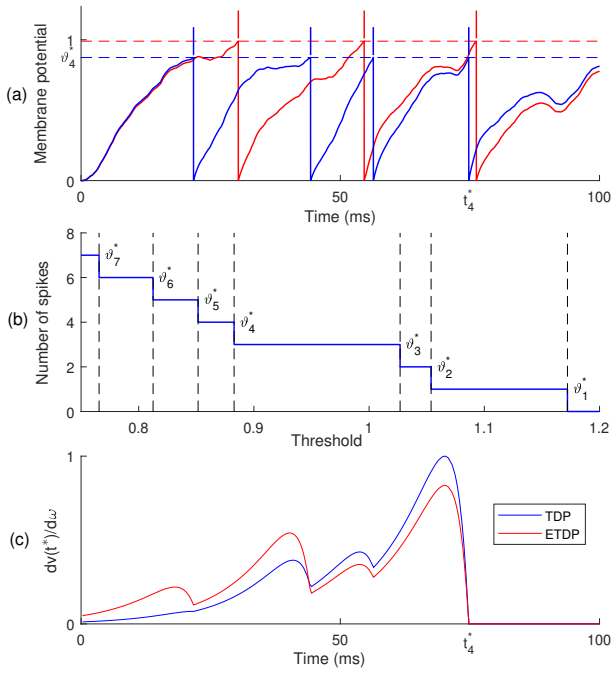

Fig. 1. (a) Membrane potential traces with the fixed biological firing threshold ϑ (red line) and the hypothetical firing threshold ϑ_4^* (blue line). (b) Illustration of the spike-threshold-surface (STS), which maps the neuron's hypothetical firing thresholds to the output spike counts. (c) The learning curve of different threshold-driven aggregate-label learning algorithms, which demonstrates the spike-timing dependence of synaptic contributions to the $dV(t^*)/d\omega$.

previous output spikes $t_s^j < t^*, j \in \{1, 2, ..., m\}$. Thus, $d\vartheta^*/d\omega_i$ can be determined as follows

$$\frac{d\vartheta^*}{d\omega_i} = \frac{dV(t^*)}{d\omega_i} = \frac{\partial V(t^*)}{\partial \omega_i} + \sum_{j=1}^{m} \frac{\partial V(t^*)}{\partial t_s^j} \frac{dt_s^j}{d\omega_i} + \frac{\partial V(t^*)}{\partial t^*} \frac{dt^*}{d\omega_i} \tag{7}$$

The last component of Eq. 7 has no contribution to the synaptic update since $V(t^*)$ is either a local maximum with $\partial V(t^*)/\partial t^* = 0$ or t^* is the time of an inhibitory input spike whose arrival time does not depend on ω_i. The difficulty in solving Eq. 7 lies in the $dt_s^j/d\omega_i$ term, and by applying the chain rule, it can be expressed as

$$\frac{dt_s^j}{d\omega_i} = \frac{\partial t_s^j}{\partial V(t_s^j)} \frac{dV(t_s^j)}{d\omega_i} \tag{8}$$

with

$$\frac{dV(t_s^j)}{d\omega_i} = \frac{\partial V(t_s^j)}{\partial \omega_i} + \sum_{k=1}^{j} \frac{\partial V(t_s^j)}{\partial t_s^k} \frac{dt_s^k}{d\omega_i} \tag{9}$$

According to the linear assumption of the firing threshold crossing [11], we get

$$\frac{\partial t_s^j}{\partial V(t_s^j)} = -\left[\frac{\partial V(t_s^j)}{\partial (t_s^j)}\right]^{-1} = -V'(t_s^j)^{-1} \tag{10}$$

Then, the Eq. 7 can be expressed as

$$\frac{d\vartheta^*}{d\omega_i} = \frac{\partial V(t^*)}{\partial \omega_i} - \sum_{j=1}^{m} \frac{\partial V(t^*)}{\partial t_s^j} \frac{1}{V'(t_s^j)} \frac{dV(t_s^j)}{d\omega_i} \tag{11}$$

In order to solve the remaining components of Eq. 11, we denote the set of output spike times as $t_x \in \{t_s^1, t_s^2, ..., t_s^m, t^*\}$. The Eq. 5 thus can be evaluated as

$$V(t_x) = \frac{V_o(t_x)}{C_{t_x}} \tag{12}$$

with

$$C_{t_x} = 1 + \sum_{t_s^j < t_x} \exp\left(-\frac{t_x - t_s^j}{\tau_m}\right) \tag{13}$$

Then, the remaining components of Eq. 11 can be determined as follows

$$\frac{\partial V(t_x)}{\partial \omega_i} = \frac{1}{C_{t_x}} \sum_{t_i^j < t_x} K(t_x - t_i^j) \tag{14}$$

$$\frac{\partial V(t_x)}{\partial t_s^k} = \frac{-V_o(t_x)}{C_{t_x}^2} \frac{\exp\left(-\frac{t_x - t_s^k}{\tau_m}\right)}{\tau_m} \qquad \text{if} \quad t_s^k < t_x \tag{15}$$

$$V'(t_s^j) = \frac{1}{C_{t_x}} \frac{\partial V_o(t_x)}{\partial t_x} + \frac{V_o(t_x)}{C_{t_x}^2 \tau_m} \sum_{t_s^j < t_x} \exp\left(-\frac{t_x - t_s^j}{\tau_m}\right) \tag{16}$$

Since the term $V'(t_s^j)$ is the denominator of Eq. 10, this will lead to a gradient explosion problem when $V'(t_s^j)$ is close to 0. To solve this problem, we propose an exploding gradient prevention strategy (EGPS) by setting a lower bound ϑ_b for $V'(t_s^j)$ as

$$V'(t_s^j) = \begin{cases} V'(t_s^j) & \text{if} \quad V'(t_s^j) > \vartheta_b \\ \vartheta_b & \text{otherwise} \end{cases} \tag{17}$$

In the same vein of research, the threshold-driven aggregate-label learning algorithm TDP simplifies the recursive expression of the MST algorithm and demonstrated significantly improved learning efficiency in their experiments [3]. Here, we focus on the difference between the proposed ETDP and the TDP algorithms. The main difference between these two algorithms lies in the different solutions to terms $\partial V(t_x)/\partial \omega_i$ and $dV(t_s^j)/d\omega_i$.

According to Eq. 5, $V(t_x)$ is defined as

$$V(t_x) = V_o(t_x) - \vartheta^* \sum_{j=1}^{m} \exp\big(-\frac{t_x - t_s^j}{\tau_m}\big) \tag{18}$$

TDP calculates $\partial V(t_x)/\partial \omega_i$ by simply considering the first term of Eq. 18 that leads to the following equation

$$\frac{\partial V(t_x)}{\partial \omega_i} = \frac{\partial V_o(t_x)}{\partial \omega_i} = \sum_{t_i^j < t_x} K(t_x - t_i^j) \tag{19}$$

However, the membrane potential $V(t_x)$ depends on the synaptic weight ω_i also through the second term of Eq. 18. To consider this dependency, the proposed ETDP rule first transforms Eq.18 into Eq. 12 following the proposal in [1], and then solves $\partial V(t_x)/\partial \omega_i$ according to Eq. 14, which is more rigorous in mathematics.

On the other hand, TDP calculates the $dV(t_s^j)/d\omega_i$ as

$$\frac{dV(t_s^j)}{d\omega_i} = \frac{\partial t_s^j}{\partial V(t_s^j)} \frac{\partial V(t_s^j)}{\partial \omega_i} \tag{20}$$

with

$$\frac{\partial V(t_s^j)}{\partial \omega_i} = \sum_{t_i^j < t_s^j} K(t_s^j - t_i^j) \tag{21}$$

which ignores the fact that the membrane potential $V(t_s^j)$ depends on the synaptic weights ω_i also through the output spikes generated before t_s^j. While we consider this dependency in the proposed ETDP rule and determine $dV(t_s^j)/d\omega_i$ as per Eq. 8 and Eq. 9. As the learning curves provided in Fig. 1(c), the ETDP will allocate more credits to the earlier presynaptic spikes compared to the TDP.

C. ETDP Learning Algorithm for Multi-layer SNNs

The existing aggregate-label learning algorithms are all based on single spiking neurons. While the powerful perceptual and cognitive capabilities of the brain come from the huge number of neurons that organized in a hierarchical manner. Therefore, these algorithms are not sufficient to simulate the learning process of biological neural networks [28][29]. Besides, the applicability of aggregate-label learning is constrained due to the limited computational capability of single spiking neurons. Therefore, in the following, we extend the proposed ETDP algorithm to multi-layer spiking neural networks.

The goal of multi-layer ETDP learning algorithm is to update the synaptic weights in both the output layer and hidden layers, such that the neurons in the output layer can generate the desired number of spikes. Same as the ETDP learning algorithm developed for single spiking neurons, this goal can be accomplished by adapting the STS such that $\vartheta_{d+1}^* < \vartheta \leq \vartheta_d^*$. Considering spiking neural networks with a single hidden layer, since the synaptic weights ω_{ih} between input layer and hidden layer affects ϑ^* through both the spikes of hidden neurons (t_h^m) and output neurons (t_j^n), the weight update rule for ω_{ih} can be expressed as

$$\frac{dV(t^*)}{d\omega_{ih}} = \sum_{t_h^m < t^*} \frac{\partial V(t^*)}{\partial t_h^m} \frac{dt_h^m}{d\omega_{ih}} + \sum_{t_j^n < t^*} \frac{\partial V(t^*)}{\partial t_j^n} \frac{dt_j^n}{d\omega_{ih}} \tag{22}$$

with

$$\frac{dt_h^m}{d\omega_{ih}} = \frac{\partial t_h^m}{\partial V(t_h^m)} \left[\frac{\partial V(t_h^m)}{\partial \omega_{ih}} + \sum_{k=1}^{m} \frac{\partial V(t_h^m)}{\partial t_h^k} \frac{dt_h^k}{d\omega_{ih}} \right] \tag{23}$$

$$\frac{dt_j^n}{d\omega_{ih}} = \frac{\partial t_j^n}{\partial V(t_j^n)} \left[\sum_{t_h^m < t_j^n} \frac{\partial V(t_j^n)}{\partial t_h^m} \frac{dt_h^m}{d\omega_{ih}} + \sum_{k=1}^{n} \frac{\partial V(t_j^n)}{\partial t_j^k} \frac{dt_j^k}{d\omega_{ih}} \right] \tag{24}$$

where t_h^m is the mth spike of the hidden neuron h, and t_j^n is the nth spike of the output neuron j. All the components in Eqs. 10, 22, 23 and 24 can be solved with a combination of Eqs. 14, 15, 16 and the following equation.

$$\frac{\partial V(t_x)}{\partial t_h^m} = \frac{\omega_{ih} V_0}{C_{t_x}} \left[\frac{1}{\tau_m} exp\left(-\frac{t_x - t_h^m}{\tau_m}\right) - \frac{1}{\tau_s} exp\left(-\frac{t_x - t_h^m}{\tau_s}\right) \right] \tag{25}$$

Furthermore, the SNNs with multiple hidden layers can be trained in a similar fashion by applying the chain rule.

III. Experimental Results

In this section, we conduct extensive experiments to evaluate the performance of the proposed ETDP learning algorithm for single spiking neurons and multi-layer SNNs. First, we evaluate the effectiveness and efficiency of the ETDP algorithm by training a single spiking neuron to generate the desired number of spikes. Then, we demonstrate that the proposed ETDP algorithm can train spiking neurons to discover useful clues embedded within a long stream of multimodal sensory activities. Finally, we evaluate the performance of the ETDP algorithm by validating on an SNN-based multimodal computational framework for audio-visual information processing.

A. Learning to Fire a Desired Number of Spikes

In this section, we first introduce a learning example to demonstrate the effectiveness of the proposed ETDP algorithm for single spiking neurons. Furthermore, the learning efficiency of this algorithm is compared with the threshold-driven aggregate-label learning algorithm TDP.

In the first set of experiments, a spiking neuron with $N = 500$ presynaptic neurons is trained to fire 10 spikes within a time window of $T = 500$ ms. The initial synaptic weights are drawn from a random Gaussian distribution with both mean and standard deviation equal to 0.01. We adjust the initial firing rate r_{pre} of presynaptic neurons to 4 and 10 Hz so as to cover both the under-firing and over-firing scenarios. The experimental results of these two scenarios are provided in Fig. 2 and Fig. 3, respectively.

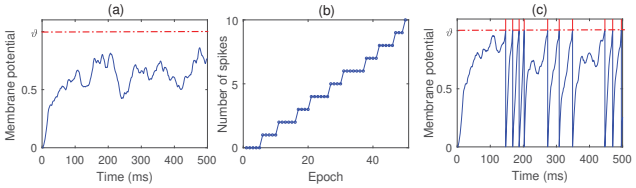

Fig. 2. Learning a desired number of spikes with $r_{pre} = 4$ Hz (under-firing scenario). (a) Neuron's membrane potential trace before learning. (b) The number of output spikes at the end of each learning epoch. (c) Neuron's membrane potential trace after learning.

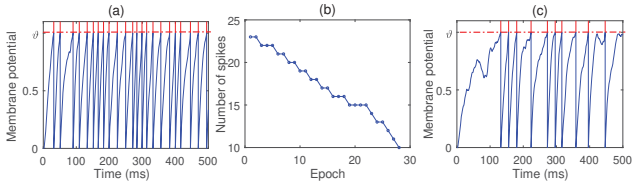

Fig. 3. Learning a desired number of spikes with $r_{pre} = 10$ Hz (over-firing scenario). (a) Neuron's membrane potential trace before learning. (b) The number of output spikes at the end of each learning epoch. (c) Neuron's membrane potential trace after learning.

Fig. 2 illustrates the learning process with an input firing rate of $r_{pre} = 4$ Hz. Due to the low input firing rate, the membrane potential of the output neuron cannot reach the firing threshold initially, and the output neuron thus remains quiescent. As shown in Fig. 2(b), when trained with the proposed ETDP learning algorithm, the output neuron gradually increases its number of output spikes and reaches the desired spike count after about 50 epochs. The membrane potential trace of a successful learning example is given in Fig. 2(c). Fig. 3 shows that the learning neuron exhibits bursting behavior with a high input rate of $r_{pre} = 10$ Hz. As learning progresses, the number of output spikes decreases to the desired spike count after 28 learning epochs. These experimental results demonstrate the proposed ETDP algorithm works effectively under different neuronal activity states.

Next, we compare the learning efficiency of the ETDP algorithm to the TDP. The experimental setup is same as that used in Fig. 2, while the desired number of spikes varies from 10 to 100 with a step of 10. For each desired spike count, 20 independent experiments are conducted, and the statistics of learning epochs and CPU time used are summarized in Fig. 4

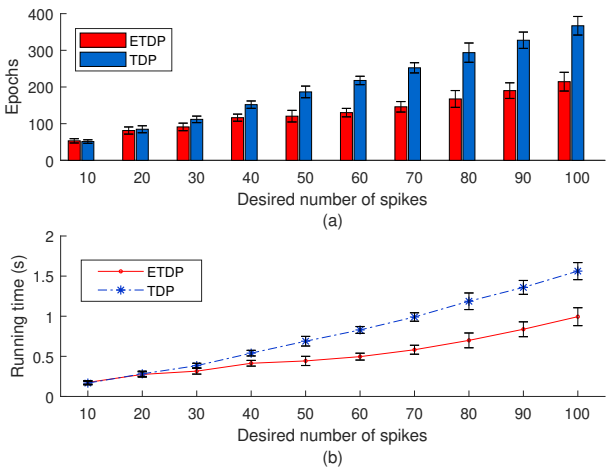

Fig. 4. Comparison of learning efficiency between the proposed ETDP and TDP. (a) The required learning epochs of different algorithms. (b) The required CPU time of different algorithms.

As shown in Fig. 4, the required number of learning epochs and CPU time increase for both learning algorithms with an increasing number of the desired spike count. However, the proposed ETDP algorithm consistently outperforms TDP for all the tasks. For example, when the desired number of spikes is 100, the required number of learning epochs of the proposed algorithm is about 200, while it is about 370 for TDP. Besides, as shown in Fig. 4(b), the required average CPU time of the ETDP algorithm is also lower than that of the TDP. Specifically, for a desired spike count of 100, the CPU time needed for our algorithm and TDP is 0.9s and 1.5s, respectively. It worth noting that despite our algorithm takes more CPU time per epoch to derive a higher quality gradient than TDP, it takes significantly shorter CPU time that is due to savings in the required training epochs.

B. Learning Multimodal Sensory Clues

Learning multimodal sensory clues can facilitate efficient identification and localization of external events, and hence enhance interactions with the environment. However, these useful clues are usually embedded within distracting streams of unrelated sensory activities, and the feedback signals may occur after long and varying delays. How to make effective

use of the aggregated feedback signals to discover useful sensory clues, known as the temporal credit-assignment (TCA) problem, remains a challening research topic for both neuroscience and machine learning. In this section, we evaluate the capability of the proposed ETDP algorithm to solve the TCA problem on both the synthetic and real-world datasets.

Similar to the tasks proposed in [1], ten brief spike patterns are constructed to represent the spiking activities in response to different multimodal sensory clues. Each brief spike pattern consists of 500 spike trains of 50 ms, wherein each spike train is generated randomly at a firing rate of 5 Hz. In each trial, as shown in Fig. 5, a random number of these ten spike patterns are embedded within a long stream of background spiking activity generated at the same firing rate of 5 Hz. Each training cycle consists of 100 such trials generated with the set-up described above. Here, the task is to enable a single spiking neuron to detect the useful sensory clues by firing the specific number of spikes during their presence.

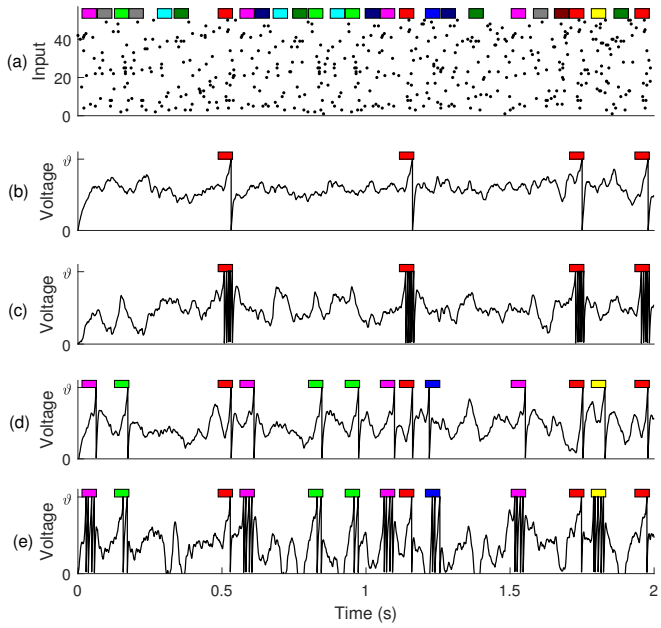

Fig. 5. Learning useful multimodal sensory clues. (a) Input spike pattern. For better visualization, only the first 50 out of the 500 afferents are provided. Colored rectangles correspond to 10 different sensory clues. (b) The learning neuron is trained to generate one spike only during the presence of the i-th clue (red rectangle). (c) The learning neuron is trained to generate a burst of five spikes only during the presence of the i-th clue. (d) The learning neuron is trained to generate one spike only during the presence of the five different clues. (e) The learning neuron is trained to generate a distinct number of spikes $\{1, 2, 3, 4, 5\}$ during the presence of the five different clues.

In Fig. 5(b), the neuron is trained to detect the clue i among the other 9 distractors and background activities. For each trial, the desired number of spikes N_d is set as the occurrences of clue i ($N_d = c_i$). If the learning neuron fires more or fewer spikes, the proposed learning algorithm will weaken or potentiate the synaptic weights to make the neuron fire desired spike count. As shown in Fig. 5(b), the learning neuron can precisely fire one spike during the presence of the clue i. As shown in Fig. 5(c), when set the desired spike count five times to the occurrences of the clue i ($N_d = 5c_i$), the neuron learns to generate a burst of 5 spikes in response to the clue i and remains silent otherwise. Moreover, by setting the desired spike count as $N_d = \sum_i c_i d_i$, where c_i denotes the number of clue i within a trial and d_i is the corresponding desired spike count to the clue i, the proposed learning algorithm enables the trained neuron to decompose the feedback signal and associate each clue with a distinct number of spikes. Fig. 5(d) and Fig. 5(e) show the testing results when d_i of the five useful sensory clues are set as $\{1, 1, 1, 1, 1\}$ and $\{1, 2, 3, 4, 5\}$, respectively. These experimental results demonstrate the proposed learning algorithm can learn useful multimodal sensory clues with delayed feedback even when these clues are embedded within distracting streams of unrelated sensory and background activities.

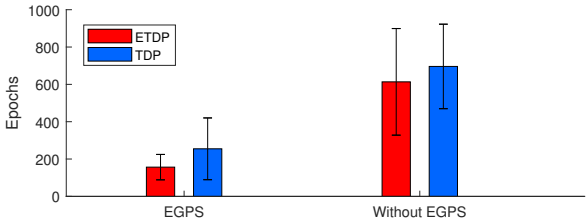

Fig. 6. Learning efficiency of different learning algorithms to accomplish the task of Fig. 5(e). The left and right figures summarize the required learning epochs with and without EGPS, respectively.

As explained in Section II-B, the derived error gradients are prone to the gradient explosion problem. Here, we evaluate the effectiveness of the proposed EGPS method to overcome this problem by comparing the required learning epochs, with and without the EGPS method, to solve the corresponding task in Fig. 5(e). As shown in Fig. 6, by combining the proposed EGPS method, the learning efficiency is improved for both the learning algorithms TDP and ETDP. Moreover, the learning efficiency of the proposed ETDP algorithm is higher than the TDP algorithm for this challenging multimodal sensory clues learning task. Specifically, when combined with the EGPS method, the required learning epochs of our method and TDP are about 150 and 250, respectively.

Next, we apply our method to a more challenging real-world task. In this task, we construct 200 multimodal spiking streams by randomly embedding 10 spike patterns, encoded from five images and five speech signals, within a long stream of background activities. These five images are randomly selected from the MNIST dataset, and further encoded into spike patterns through the latency coding [4][31] as illustrated in Fig. 7. These five speech signals are randomly selected from the TIDIGITS corpus, and then encoded into spike patterns using the Biologically plausible Auditory Encoding scheme (BAE) [32][30] as shown in Fig. 8. There are two neurons in the output layer, which selectively respond to images and speech signals, respectively. The desired spike count of each output neuron is defined as $N_d = \sum_i c_i d_i$, where c_i denotes the number of clue i (i-th image or i-th speech signal) within a spiking stream, and d_i is the corresponding desired spike count of the clue i.

After training, we generate a testing spike stream to verify whether these two output neurons can separate and recog-

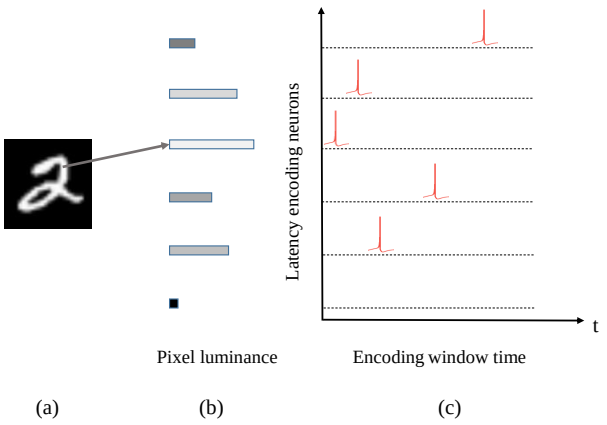

Fig. 7. The illustration of the neural latency coding for images. The luminance or intensity value of each pixel is encoded into the spike time, whereby the earlier spike time corresponds to the larger intensity value. (a) is an image of the hand-written digit "2". The horizontal bars in (b) depict the luminance or intensity value of 6 pixels, where a longer bar represents a brighter pixel. (c) is the latency-encoded spike pattern, in which each pixel in (b) is encoded into a single spike (red pulse) in the corresponding row of (c).

nize different visual and auditory clues. Fig. 9(b) and Fig. 9(c) illustrate the membrane potential traces of the neurons that trained to selectively respond to auditory (speech) and visual (image) information, respectively. After training with the proposed ETDP learning algorithm, these two output neurons can selectively respond to speech signals and images. Furthermore, they can recognize different clues by firing the corresponding number of spikes. For example, as shown in Fig. 9(c), this output neuron fires spikes whenever there is an image presented, while remains silent during the presence of speech signals and background activities. Besides, the neuron fires a distinct number of spikes in response to different images.

C. Classification Tasks

To demonstrate the effectiveness of the proposed ETDP learning algorithm for multi-layer SNNs, we first validate the trained SNNs on the XOR classification task. Furthermore, we propose an SNN-based computational framework for multimodal pattern recognition tasks.

1) XOR Classification Task: In this experiment, we encode the four training samples of the XOR task into spike time by associating the binary input '0' and '1' to spike times of 5 ms and 10 ms, respectively. The input spikes then project to a hidden layer consists of four neurons which subsequently connected to a single output neuron. During the training process, the training samples of $\{5, 5\}$ ms and $\{10, 10\}$ ms are defined as the same class, and the output neuron is required to fire two spikes. While when the samples of $\{5, 10\}$ ms and $\{10, 5\}$ ms are presented to the network, the output neuron is required to remain silent.

As shown in Fig. 10(a), there are four different input spike patterns corresponding to the four training samples. Fig. 10(b) shows the membrane potential traces of the four hidden spiking neurons which are denoted in different colors. After training, the output neuron can precisely emit two spikes when the samples of $\{5, 5\}$ ms and $\{10, 10\}$ ms are presented, while remains silent otherwise. This experimental result suggests that the proposed ETDP learning algorithm has the capability to trained multi-layer SNNs to perform the non-linear pattern classification task.

2) Multimodal Pattern Recognition: The studies in cognitive neuroscience suggest that the human brain can efficiently integrate sensory information of multiple modalities [33], [34], [35], [36]. Besides, there is strong evidence showing that cross-modal coupling facilitates the influence of one modality to the areas of other modalities, and the integration occurs in the supramodal areas where neurons are sensitive to multimodal stimuli. Inspired by these findings, we propose an SNN-based multimodal computational framework for audio-visual pattern recognition. As shown in Fig. 11, the proposed multimodal computational frame mainly consists of three parts, the unimodal processing part, the cross-modal coupling part, and the supramodal part. In the following, the working mechanism of each part will be introduced in sequence.

In the unimodal processing part, two SNN-based computational models are working independently for visual and audio modalities. These two unimodal SNN models are trained following the proposed ETDP algorithm. The feedforward SNN architectures used for visual and audio signal processing are 784-800-10 and 620-800-10, respectively. The role of cross-modal coupling is to transmit the influence of one modality to the areas that intrinsically belong to other modalities. Hence, in the cross-modal coupling part, we construct excitatory and inhibitory connections across two different modalities. Such that when the output neurons of one modality fire spikes, the output neurons in the other modality will receive those spikes to facilitate synchronized behaviors across different modality. For example, when both the image and speech patterns 'one' are presented to the unimodal SNNs, and the output neuron representing image 'one' fires first. The generated spikes will excite the output neuron representing 'one' of the audio modality, while inhibit all other neurons to prevent them from firing.

There are ten neurons in the supramodal layer, which integrate the information from the corresponding output neurons of single modalities through excitatory connections. To facilitate a rapid response, the neurons in the supramodal layer will generate an output spike as soon as they receive an incoming spike from the cross-modal coupling layer.

We evaluate the performance of the proposed multimodal computational framework on the joint digit classification dataset. In this experiment, the training dataset consists of 60,000 pairs of inputs (training samples in the TIDIGITS corpus are repeated to match the size of the MNIST dataset), and the testing dataset consists of 10,000 samples. Same as the earlier experiments, we use the latency coding [4][31] and the Biologically plausible Auditory Encoding Scheme [30][32] to encode the image and speech signals into spike patterns, respectively. When the encoded spike pattern is presented to the unimodal SNN, the corresponding output neuron is trained with the proposed ETDP learning algorithm such that it fires the most number of spikes. The connections between different modalities are pre-defined so as to exert the desired influence on the other modality. In the supramodal part, the pattern is

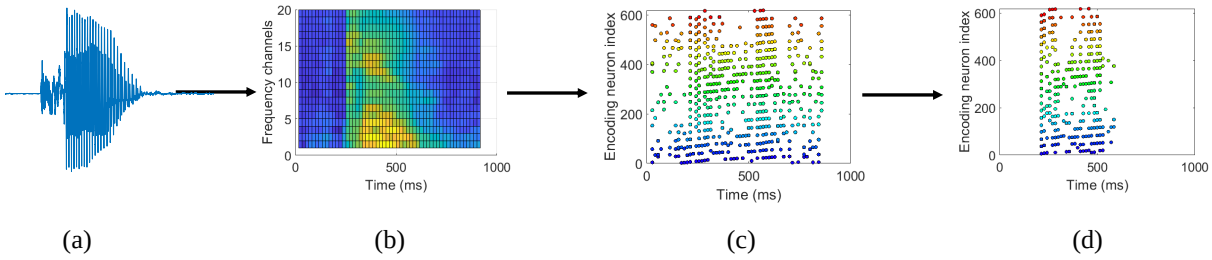

Fig. 8. The illustration of the neural encoding for audio signals, using a Biologically plausible Auditory Encoding scheme (BAE). A raw audio signal corresponds to the spoken digit "two" (a) is first filtered by a cochlear filter bank and decomposed into a 20-channel spectrogram (b). We further encode this spectrogram with the neural threshold coding (c), which can effectively describe the moving trajectory of sub-band signals. Finally, we apply an auditory masking scheme to eliminate those imperceptible spikes, resulting in a sparse while effective spike pattern (d). More details about the BAE scheme can be found in [30].

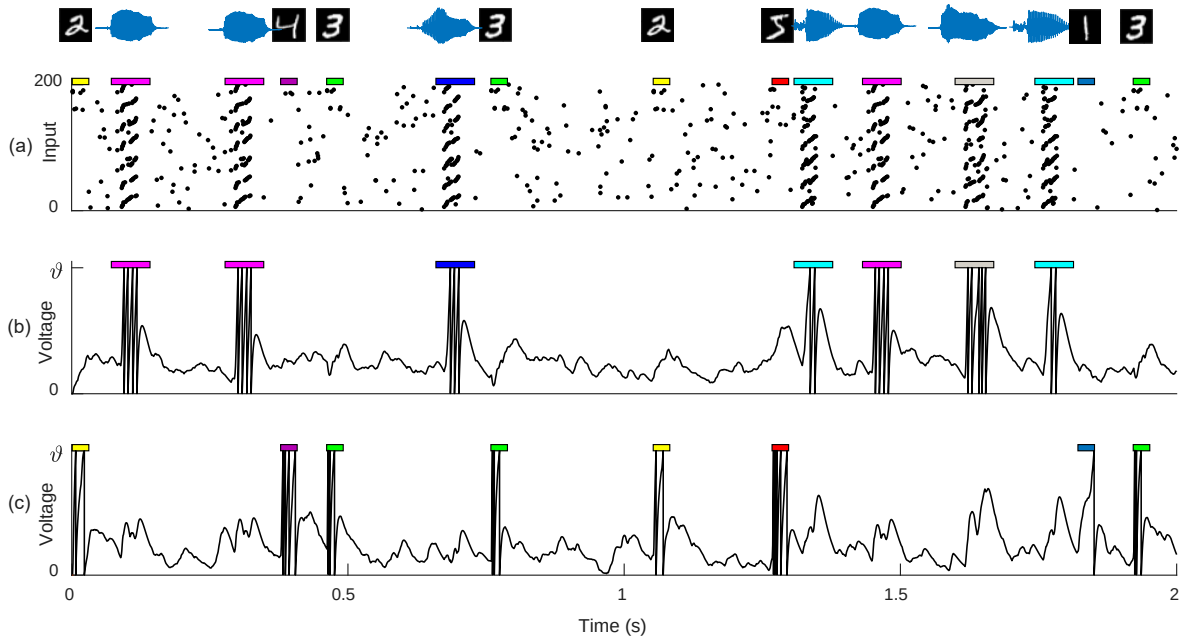

Fig. 9. Illustration of the audio-visual pattern recognition with spiking neurons. (a) The input spiking stream corresponds to the audio-visual sensory stimuli on the top row, the random spontaneous spiking activities are added during the silence period. Only the first 200 synaptic afferents are given. (b) The membrane potential trace of the output neuron that is trained to selectively respond to speech signals. (c) The membrane potential trace of the output neuron that is trained to selectively respond to images.

classified to the neuron that fires the most number of spikes.

As shown in Table. I, the multimodal classification framework equipped with the proposed ETDP learning algorithm outperforms many unimodal approaches. In addition, with the help of crossmodal coupling and the supramodal parts, the multimodal classification framework achieves a classification accuracy of 98.9%, which improves over single modalities by more than 2%.

IV. Discussion

The aggregate-label learning paradigm equips spiking neurons with the capability to decompose the aggregated supervision signals into both spatial and temporal domains, whereby effectively solves the long-standing 'temporary credit assignment' problem in neuroscience. Comparing with other existing SNN learning algorithms[37], [43], [44], [45], [46], the aggregate-label learning paradigm boosts the computational capability of a single spiking neuron by making it fire a distinct number of spikes in response to different predictive clues.

The existing aggregate-label learning algorithms can be classified into membrane potential-driven and threshold-driven methods. For membrane potential-driven methods, the synaptic updates are directly derived from the subthreshold membrane potentials. While the threshold-driven methods construct a spike-threshold-surface to map the discrete spike counts to the continuous hypothetical firing thresholds and perform synaptic updates based on the error gradients derived from the spike-threshold-surface. By avoiding the computational-intensive process of calculating the hypothetical threshold ϑ^*, the efficiency of membrane potential-driven methods is significantly improved over their threshold-driven counterparts.

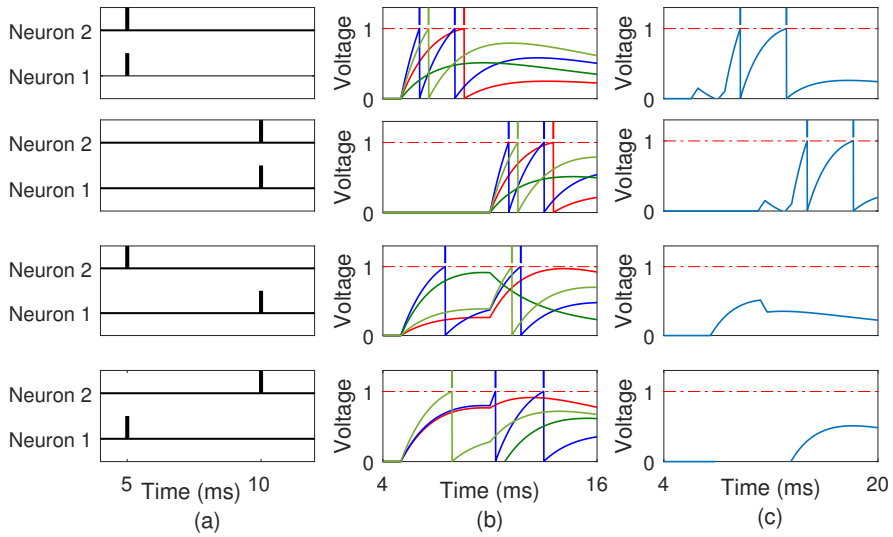

Fig. 10. Illustration of the XOR classification task with the multi-layer SNN. (a) Four input spike patterns are constructed by associating the binary input '0' and '1' to spike times of 5 ms and 10 ms, respectively. (b) The membrane potential traces of the four hidden neurons after training. The membrane potential traces are color-coded to denote different hidden neurons. (c) The membrane potential traces of the output neuron corresponding to different input spike patterns.

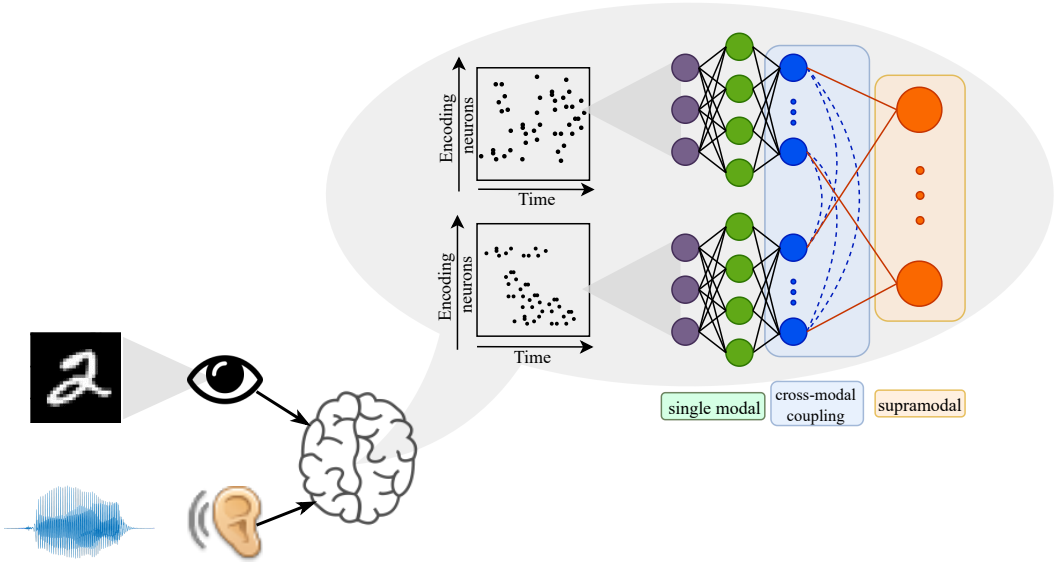

Fig. 11. The proposed SNN-based computational framework for multimodal pattern recognition. This framework mainly consists of three parts, the single modal processing part, the cross-modal coupling part and the supramodal part.

However, the membrane potential-driven methods, such as MPD-AL, are subject to several limitations. First of all, the synaptic updates of the MPD-AL algorithm are dependent on the availability of a maximum peak in the subthreshold membrane potential trace. Whenever there is no such a peak exist in between any two adjacent spike times, the learning process is stopped. Furthermore, the membrane potential-driven methods can learn predictive clues only when they are sparsely embedded in training samples [2]. In contrast, the threshold-driven algorithms are not constrained by the existence of the maximum peak or the sparsity of embedded clues.

The proposed ETDP learning algorithm improves the learning efficiency over other existing membrane potential-driven methods by optimizing the learning curve and preventing the problem of gradient explosion. As demonstrated in our experiments, the required training epochs and CPU time are improved consistently across different pattern recognition tasks. While it is worth noting that the calculation of ϑ^* is still time-consuming for all the threshold-driven methods, we will explore efficient strategies to calculate this quantity in our future work. The existing aggregate-label learning algorithms can only train single spiking neurons to output a desired number of spikes. However, the powerful perceptual and cog-

TABLE I
COMPARISON OF OUR WORK WITH OTHER UNIMODAL APPROACHES

Model	Type	Layers	Learning	Modality	Dataset	Accuracy
Diehl et al. [37]	SNN	2	Unsupervised	Unimodal	MNIST	95.0%
Rathi et al. [36]	SNN	3	Unsupervised	Unimodal	MNIST	93.2%
Kheradpisheh et al. [38]	SNN+SVM	6	Supervised	Unimodal	MNIST	98.4%
Hong et al. [39]	SNN	3	Supervised	Unimodal	MNIST	97.2%
Gu et al. [27]	SNN	3	Supervised	Unimodal	MNIST	98.6%
Tavanaei et al. [40]	SNN+SVM	2	Supervised	Unimodal	TIDIGITS	91.0%
Tavanaei et al. [41]	SNN+HMM	4	Supervised	Unimodal	TIDIGITS	96.0%
Neil et al. [42]	MFCC and RNN	4	Supervised	Unimodal	TIDIGITS	96.1%
ETDP (this work)	SNN	3	Supervised	Unimodal	MNIST	**96.8%**
ETDP (this work)	SNN	3	Supervised	Unimodal	TIDIGITS	**95.8%**
ETDP (this work)	SNN	3	Supervised	Multimodal	MNIST and TIDIGITS	**98.9%**

nitive capabilities of cortical neural networks are accomplished with a large number of biological neurons that are organized hierarchically. In this paper, for the first time, we introduce an aggregate-label learning algorithm for multi-layer SNNs by combining the proposed ETDP algorithm with the spike-based error back-propagation.

We further develop an SNN-based multimodal computational framework that can effectively integrate sensory information from multiple modalities for effective decision making. This framework consists of the unimodal processing units, the cross-modal coupling part, and the supramodal part. It is worth noting that the cross-modal coupling part facilitates the information synchronization across unimodal processing units that handling different sensory modalities. Finally, the supramodal part effectively integrates the information of different sensory modalities and significantly improves the decision quality as demonstrated in the digit recognition task.

V. CONCLUSION

The temporal credit assignment problem is a long-standing research topic in neuroscience and machine learning. In this work, we propose an efficient threshold-driven aggregate-label learning algorithm, namely ETDP, to resolve this challenging problem. The ETDP algorithm optimizes the learning curve over the existing threshold-driven aggregate-label learning algorithms, thereby achieves significantly improved learning efficiency and effectiveness. Furthermore, we extend the ETDP algorithm to support multi-layer network configurations. To the best of our knowledge, this is the first time that an aggregate-label learning algorithm is developed for multi-layer SNNs. Finally, we propose an SNN-based computational framework for multimodal sensory information processing. Equipped with the proposed ETDP algorithm, this framework achieves superior classification accuracy over other unimodal frameworks. As future work, we will apply the ETDP algorithm to convolutional SNNs so as to better process the visual information and explore more challenging multimodal sensory information processing tasks.

.

REFERENCES

[1] R. Gütig, "Spiking neurons can discover predictive features by aggregate-label learning," *Science*, vol. 351, no. 6277, p. aab4113, 2016.

[2] M. Zhang, J. Wu, Y. Chua, X. Luo, Z. Pan, D. Liu, and H. Li, "Mpd-al: an efficient membrane potential driven aggregate-label learning algorithm for spiking neurons," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 1327–1334.

[3] Q. Yu, H. Li, and K. C. Tan, "Spike timing or rate? neurons learn to make decisions for both through threshold-driven plasticity," *IEEE Transactions on Cybernetics*, 2018.

[4] J. J. Hopfield, "Pattern recognition computation using action potential timing for stimulus representation," *Nature*, vol. 376, no. 6535, p. 33, 1995.

[5] W. Gerstner and W. M. Kistler, *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge University Press, 2002.

[6] R. Gütig and H. Sompolinsky, "The tempotron: a neuron that learns spike timing–based decisions," *Nature Neuroscience*, vol. 9, no. 3, p. 420, 2006.

[7] M. Zhang, H. Qu, A. Belatreche, Y. Chen, and Z. Yi, "A highly effective and robust membrane potential-driven supervised learning method for spiking neurons," *IEEE Transactions on Neural Networks and Learning Systems*, 2018.

[8] S. Thorpe, A. Delorme, and R. Van Rullen, "Spike-based strategies for rapid processing," *Neural Networks*, vol. 14, no. 6-7, pp. 715–725, 2001.

[9] N. Kasabov, K. Dhoble, N. Nuntalid, and G. Indiveri, "Dynamic evolving spiking neural networks for on-line spatio-and spectro-temporal pattern recognition," *Neural Networks*, vol. 41, pp. 188–201, 2013.

[10] J. Wang, A. Belatreche, L. P. Maguire, and T. M. McGinnity, "Spiketemp: An enhanced rank-order-based learning approach for spiking neural networks with adaptive structure," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 1, pp. 30–43, 2015.

[11] S. M. Bohte, J. N. Kok, and H. La Poutre, "Error-backpropagation in temporally encoded networks of spiking neurons," *Neurocomputing*, vol. 48, no. 1-4, pp. 17–37, 2002.

[12] B. Zhao, R. Ding, S. Chen, B. Linares-Barranco, and H. Tang, "Feedforward categorization on aer motion events using cortex-like features in a spiking neural network," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 9, pp. 1963–1978, 2014.

[13] Q. Yu, H. Tang, K. C. Tan, and H. Li, "Rapid feedforward computation by temporal encoding and learning with spiking neurons," *IEEE transactions on neural networks and learning systems*, vol. 24, no. 10, pp. 1539–1552, 2013.

[14] J. Wu, Y. Chua, M. Zhang, H. Li, and K. C. Tan, "A spiking neural network framework for robust sound classification," *Frontiers in neuroscience*, vol. 12, 2018.

[15] F. Ponulak and A. Kasiński, "Supervised learning in spiking neural networks with resume: sequence learning, classification, and spike shifting," *Neural Computation*, vol. 22, no. 2, pp. 467–510, 2010.

[16] A. Taherkhani, A. Belatreche, Y. Li, and L. P. Maguire, "Dl-resume: A delay learning-based remote supervised method for spiking neurons," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 12, pp. 3137–3149, 2015.

[17] R. V. Florian, "The chronotron: a neuron that learns to fire temporally precise spike patterns," *PloS one*, vol. 7, no. 8, p. e40233, 2012.

[18] A. Mohemmed, S. Schliebs, S. Matsuda, and N. Kasabov, "Span: Spike pattern association neuron for learning spatio-temporal spike patterns," *International Journal of Neural Systems*, vol. 22, no. 04, p. 1250012, 2012.

[19] J. D. Victor and K. P. Purpura, "Metric-space analysis of spike trains: theory, algorithms and application," *Network: computation in neural systems*, vol. 8, no. 2, pp. 127–164, 1997.

[20] M. v. Rossum, "A novel spike distance," *Neural computation*, vol. 13, no. 4, pp. 751–763, 2001.

[21] M. Zhang, H. Qu, A. Belatreche, and X. Xie, "Empd: An efficient membrane potential driven supervised learning algorithm for spiking neurons," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 2, pp. 151–162, 2018.

[22] Y. Xu, X. Zeng, and S. Zhong, "A new supervised learning algorithm for spiking neurons," *Neural Computation*, vol. 25, no. 6, pp. 1472–1511, 2013.

[23] R.-M. Memmesheimer, R. Rubin, B. P. Ölveczky, and H. Sompolinsky, "Learning precisely timed spikes," *Neuron*, vol. 82, no. 4, pp. 925–938, 2014.

[24] X. Luo, H. Qu, Y. Zhang, and Y. Chen, "First error-based supervised learning algorithm for spiking neural networks," *Frontiers in Neuroscience*, vol. 13, 2019.

[25] Q. Yu, L. Wang, and J. Dang, "Neuronal classifier for both rate and timing-based spike patterns," in *International Conference on Neural Information Processing*. Springer, 2017, pp. 759–766.

[26] R. Xiao, Q. Yu, R. Yan, and H. Tang, "Fast and accurate classification with a multi-spike learning algorithm for spiking neurons," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 1445–1451. [Online]. Available: https://doi.org/10.24963/ijcai.2019/200

[27] P. Gu, R. Xiao, G. Pan, and H. Tang, "Stca: Spatio-temporal credit assignment with delayed feedback in deep spiking neural networks," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 1366–1372. [Online]. Available: https://doi.org/10.24963/ijcai.2019/189

[28] Y. Xu, X. Zeng, L. Han, and J. Yang, "A supervised multi-spike learning algorithm based on gradient descent for spiking neural networks," *Neural Networks*, vol. 43, pp. 99–113, 2013.

[29] I. Sporea and A. Grüning, "Supervised learning in multilayer spiking neural networks," *Neural computation*, vol. 25, no. 2, pp. 473–509, 2013.

[30] Z. Pan, Y. Chua, J. Wu, M. Zhang, H. Li, and E. Ambikairajah, "An efficient and perceptually motivated auditory neural encoding and decoding algorithm for spiking neural networks," *arXiv preprint arXiv:1909.01302*, 2019.

[31] J. Hu, H. Tang, K. C. Tan, H. Li, and L. Shi, "A spike-timing-based integrated model for pattern recognition," *Neural computation*, vol. 25, no. 2, pp. 450–472, 2013.

[32] R. Gütig and H. Sompolinsky, "Time-warp–invariant neuronal processing," *PLoS Biology*, vol. 7, no. 7, p. e1000141, 2009.

[33] G. A. Calvert, "Crossmodal processing in the human brain: insights from functional neuroimaging studies," *Cesrebral cortex*, vol. 11, no. 12, pp. 1110–1123, 2001.

[34] K. v. Kriegstein, A. Kleinschmidt, P. Sterzer, and A.-L. Giraud, "Interaction of face and voice areas during speaker recognition," *Journal of cognitive neuroscience*, vol. 17, no. 3, pp. 367–376, 2005.

[35] S. G. Wysoski, L. Benuskova, and N. Kasabov, "Evolving spiking neural networks for audiovisual information processing," *Neural Networks*, vol. 23, no. 7, pp. 819–835, 2010.

[36] N. Rathi and K. Roy, "Stdp-based unsupervised multimodal learning with cross-modal processing in spiking neural network," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2018.

[37] P. U. Diehl and M. Cook, "Unsupervised learning of digit recognition using spike-timing-dependent plasticity," *Frontiers in computational neuroscience*, vol. 9, p. 99, 2015.

[38] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, "Stdp-based spiking deep convolutional neural networks for object recognition," *Neural Networks*, vol. 99, pp. 56–67, 2018.

[39] C. Hong, X. Wei, J. Wang, B. Deng, H. Yu, and Y. Che, "Training spiking neural networks for cognitive tasks: A versatile framework compatible with various temporal codes," *IEEE transactions on neural networks and learning systems*, 2019.

[40] A. Tavanaei and A. S. Maida, "A spiking network that learns to extract spike signatures from speech signals," *Neurocomputing*, vol. 240, pp. 191–199, 2017.

[41] A. Tavanaei and A. Maida, "Bio-inspired multi-layer spiking neural network extracts discriminative features from speech signals," in *International Conference on Neural Information Processing*. Springer, 2017, pp. 899–908.

[42] D. Neil and S.-C. Liu, "Effective sensor fusion with event-based sensors and deep network architectures," in *Circuits and Systems (ISCAS), 2016 IEEE International Symposium on*. IEEE, 2016, pp. 2282–2285.

[43] S. M. Bohte, H. La Poutré, and J. N. Kok, "Unsupervised clustering with spiking neurons by sparse temporal coding and multilayer rbf networks," *IEEE Transactions on neural networks*, vol. 13, no. 2, pp. 426–435, 2002.

[44] Y. Wu, L. Deng, G. Li, J. Zhu, Y. Xie, and L. Shi, "Direct training for spiking neural networks: Faster, larger, better," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 1311–1318.

[45] J. Wu, Y. Chua, M. Zhang, G. Li, H. Li, and K. C. Tan, "A hybrid learning rule for efficient and rapid inference with spiking neural networks," *arXiv preprint arXiv:1907.01167*, 2019.

[46] J. Wu, Y. Chua, M. Zhang, Q. Yang, G. Li, and H. Li, "Deep spiking neural network with spike count based learning rule," *arXiv preprint arXiv:1902.05705*, 2019.