MSAT: Multi-stage adaptive threshold for Deep Spiking Neural Networks

David S. Hippocampus*
Department of Computer Science
Cranberry-Lemon University
Pittsburgh, PA 15213
hippo@cs.cranberry-lemon.edu

Abstract

Spiking Neural Networks(SNNs) can do inference with low-power consumption natively because of its spike sparsity. Compared with the other two traning method: STDP and BP, Conversion from Artifical Neural Networks(ANNs), is a more easlier way to acheive deep SNNs and commly have the approximate performance compared with ANN. However, Conversion SNNs suffer from a accuracy degration and more latency at inference time. Lots of studys have tryed to make a trade-off between improving accuracy and reduce the latency using varied method including adjust ANN topology when mapping ANN to SNN, using a more efficient fring mechanism et.al Here we analyze conversion loss layer-to-layer and point it out that membrane potential matters in both SNN accuracy and inference latency. subsequently, we give a new perspective that most of current conversion method is optimization membrane potential to achieve higher accuracy and short latency. Meanwhile, Different from current converson schemes which use the same and invariant threshold with inference time in a layer, we propose a multi-stage adaptive threshold for deep spiking neural Networks. We examine the performance on CIFAR-100 and ImageNet for classfication. Futuremore, we show the propose method also behave well in objection detection on VOC and COCO. All above provide support on biological interpretability.

1 Introduction

At present, Artificial Neural Network (ANN) is widely used in speech recognition, image processing and other fields. However, with the complexity of neural networks increasing progressively, running such deep networks often requires large amounts of computational resources, such as memory and power. In addition, current ANN's work mechanism differs from out brain. Actually, Neurons in the brain communicate by transmitting sequences (i.e. spike) generated by action potentials. Spiking Neuron network (SNN) works in a similar way. It also transmits the spike sequence to the downstream neurons. These spikes often carry a high amount of information, and the spike distribution is sparse, so it has the characteristics of low power consumption.

SNNs potentially offer an efficient way of doing inference when it conbines with neuroncomputing hardware, futuremore, SNN inherently shows efficiency on processing temporal and spatial data. Its diverse coding mechanisms, and events-driven characteristics are also promising. However, because the internal state variables of neurons do not satisfy the continuously differentiable requirement, it is difficult to be trained. To solve this problem, some algorithms based on the rules of gradient descent and spike-time dependent plasticity (STDP) were proposed, which had partly solved the problem

*Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

Preprint. Under review.

of training SNNs. Frustratingly, It is still difficult to train deeper SNNs with complex network structures, and results in a remaining of huge gap of performance between SNNs and CNNs in complex recognition or detection tasks.

To narrow the performance gap between SNNs and CNNs, methods of converting CNNs to SNNs had been proposed. In these methods, a CNN is firstly trained using the standard stochastic gradient descent and back propagation algorithm, and then the trained weights are mapped to an SNN with the same structure as the CNN. Inference is performed on the converted SNNs. The main idea is that the firing rates of spiking neurons can approximate the activations of their counterparts (RELU) in ANNs with sufficient time steps. This finding has become the fundamental principle underlying the conversion scheme. Converted SNNs often suffer from a accu- racy degration and more latency at inference time. Lots of studys have tryed to make a trade-off between improving accuracy and reduce the latency using var- ied method including adjust ANN topology when mapping ANN to SNN, using a more efficient fring mechanism et.al. Here we forms conversion loss formula and shows that residual membrane potential in each IF neuron increase the latency which mean firing rates approximate to activation value. We also find that most of current converson schemes, they use threshold invariant with inference time and are same and in a layer. This mechanism is inconsistent with a phenomenon which has been widely observed in the central nervous system, e.g. visual cortex , auditory midbrain, hippocampus, somatosensory cortex. It has been proposed that threshold variability reflects an adaptation of the spike threshold to the membrane potential. Inspired by this, we propose a multi-stage adaptive threshold for deep spiking neural Networks. For each neuron, its threshold vaires with its own membrane potential.We both do experimental on object recognition and detection in non-trival datasets to prove proposed method is as well as efficiency with the current mainstream schemes when doing visual tasks.

Our major contribution can be summarized as:

- sufficient experimental on object recognition and detection in non-trival datasets, shows that our proposed method is both efficiency and biological interpretability
- a formula on layer-by-layer conversion error, a new perspective diving existing method into three part
- a multi-stage adaptive threshold mechanism, which is widely existing in the center nervous system and thus more biological plausible. We use it for deep spiking neural Networks.

2 PRELIMINARIES

Our conversion pipeline exploits the threshold balancing mechanism (Diehl et al., 2015; Sengupta et al., 2018) between ANN and SNN with modified ReLU function on the source ANN to reduce the consequential conversion error. Through in this mechanism, we give a two-layer MLP conversion Framework Diagram as fig1 to Convenient for our discussion.

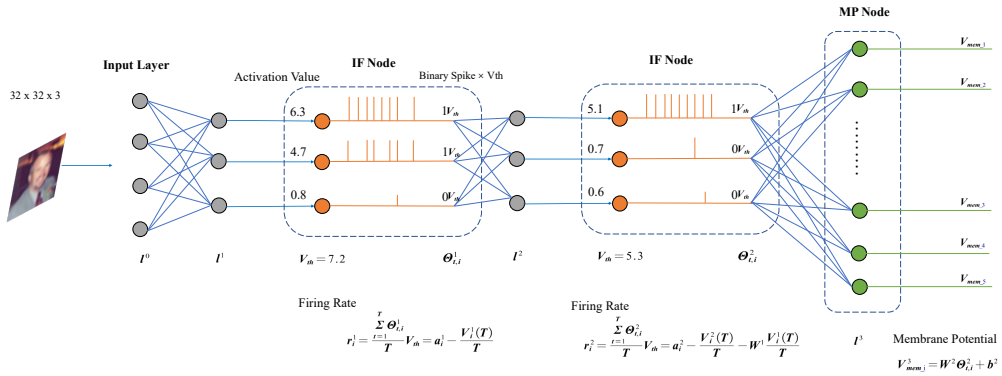


Figure 1: A two-layer MLP conversion Framework for demonstration.

The main idea in ANN-to-SNN conversion is using mean firing rate $r_i^l(t)$ which indicates firing rate of neuron i in layer l for a total time t to approximate the activation value a_i^l in SNN. Here we give analytical explanation for the approximation.

In ANN, the neuron i activation value(after relu) in layer l a_i^l can be computed as:

$$a_i^l = max\left(0, \sum_{j=1}^{M^{l-1}} W_{ij}^l a_j^{l-1} + b_i^l\right) \tag{1}$$

here $l \in \{1, \cdots, L\}$ indicates layer l in a network with L layers; W_{ij}^l indicates weight connection between neuron i in layer l and neuron j in layer $l-1$; b_i^l indicates neuron i bias in layer l; it is worth noting that a_i^l start from $l = 0$ and $a^0 = x$.

Neuron Model postsynaptic membrane potential(PSP) at timestep $t+1$, $V_i^l(t+1)$ is a sum of last timestep membrane potential and current input electric current. When PSP exceeds a certain voltage threshold, it emits an output spike and reset the membrane potential. One of the most widely adopted model is Integrate-and-Fire (IF) neuron, and membrane potential at the next time step t + 1 would then be updated by soft-reset mechanism, which subtract threshold in PSP rather than reset the membrane potential to V_{reset}. The mathematical form is as follows.

$$V_i^l(t+1) = V_i^l(t) + V_{th}^l\left(\sum_j^{M^{l-1}} W_{ij}^l \Theta_{t,j}^{l-1} + b_i^l\right) - V_{th}^l \Theta_{t,i}^l \tag{2}$$

here $\Theta_{t,i}^l$ is a function indicating the neuron i in layer l occurrence of a spike at time t

$$\Theta_{t,i}^l = \Theta\left(V_i^l(t-1) + z_i^l(t) - V_{th}^l\right), with\ \Theta(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \tag{3}$$

Here $z_i^l(t)$ is neuron i inputs in layer l and time t

$$z_i^l(t) = V_{th}^l\left(\sum_j^{M^{l-1}} W_{ij}^l \Theta_{t,i}^l + b_i^l\right) \tag{4}$$

3 Diving Conversion Error

Error comes from two part: one is converting ANN to SNN directly, result quantization error and clip error; the other is spike arrtribute, result residual potential error. show as fig 2

3.1 Layer-By-Layer Quantization Error

For equation (2), cumulate the input over the simulation timestep T, we can derive the firing rate relationship layer-to-layer.

$$V_{th}^l \sum_{t=0}^{T} \Theta_{t,i}^l = \sum_{t=0}^{T} V_{th}^l\left(\sum_j^{M^{l-1}} W_{ij}^l \Theta_{t,j}^{l-1} + b_i^l\right) - V_i^l(T) \tag{5}$$

$$r_i^l(T) = \sum_j^{M^{l-1}} W_{ij}^l r_j^{l-1} + b_i^l - \frac{V_i^l(T)}{T V_{th}^l} \tag{6}$$

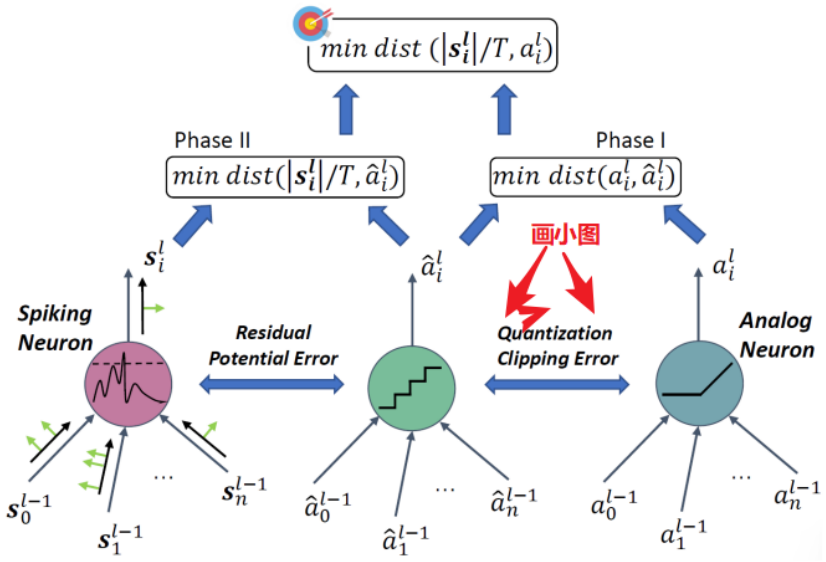


Figure 2: Three error.

Unfolding for each layer, relationship between mean fring rate and activation value can be shown as following:

$$
\begin{aligned}
r_i^l(T) &= \sum_j^{M^{l-1}} W_{ij}^l r_j^{l-1} + b_i^l - \frac{V_i^l(T)}{TV_{th}^l} \\
&= \sum_j^{l-1} \sum_j^{M^{l-1}} W_{ij}^l \left(\sum_k^{M^{l-2}} W_{jk}^{l-1} r_k^{l-2} + b_j^{l-1} - \frac{V_j^{l-1}(T)}{TV_{th}^{l-1}} \right) + b_i^l - \frac{V_i^l(T)}{TV_{th}^l} \\
&= \sum_j^{l-1} \sum_j^{M^{l-1}} W_{ij}^l \left(\sum_k^{M^{l-2}} W_{jk}^{l-1} \left(\sum \cdots \underbrace{\sum_m^{M^1} W_{1m}^1 x_m + b_m^1}_{a_m^1} - \frac{V_i^1(T)}{TV_{th}^1} + \cdots + b - \frac{V(T)}{TV_{th}} \right) \right. \\
&\quad \left. + b_j^{l-1} - \frac{V_j^{l-1}(T)}{TV_{th}^{l-1}} \right) + b_i^l - \frac{V_i^l(T)}{TV_{th}^l}
\end{aligned}
\tag{7}
$$

Use ΔV_i^l denotes $\frac{V_i^l(T)}{TV_{th}^l}$

$$
r_i^l(T) = a_i^l - \Delta V_i^l - \sum_j^{M^{l-1}} W_{ij} \Delta V_j^l - \cdots - \sum_j^{M^{l-1}} W_{ij}^l \cdots \sum_k^{M^1} W_{1k}^2 \Delta V_k^1 \tag{8}
$$

Note that the activation value is strictly fall in [0, 1] by using weight normalization. So the weights are not origin ANN weight and are scaled by V_{th}^{l-1}/V_{th}^l and bias are scaled by V_{th}^l and V_{th} is set to 1. It has the same form with threshold balancing, the different is that threshold balancing use postsynaptic neuron threshold times firing spike to compute mean firing rate Actually, Two method: weight normalization and threshold balancing are mathematically equivalent. We use threshold balancing in rest of paper for convenience.

Bias is a constant all the time so it does't affect the conversion error and we omit it and in threshold balancing, mean firing rate $r_i^l(t)$ is PSP average value, so equation (5) becomes

$$r_i^l(T) = \sum_j^{M^{l-1}} W_{ij}^l r_j^{l-1} - \frac{V_i^l(T)}{T} \tag{9}$$

When V_{th}^l is larger than maximum of activation value, $V_i^l(T)$ will be less than V_{th} thus the residual membrane potential cannot be output thats why information transmition suffer a loss. There error is basically because the discrete of timestep that the mean firing rate is a step function which cannot exactly approximate the source continuous RELU function, which is known as quantization error (flooring error), is can be expressed as

$$r_i^l(T) = \text{clip}\left(\frac{V_{th}^l}{T}\left\lfloor\frac{W_{ij}^l r_j^{l-1} T}{V_{th}^l}\right\rfloor, 0, V_{th}\right) \tag{10}$$

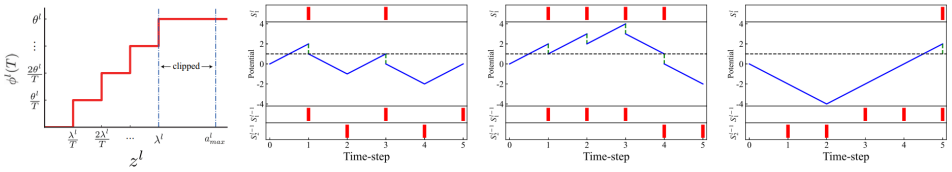
3.2 Maximum Activation Clip Error

As mentioned in equation (9), if voltage threshold is set less than maximum activation value, then when PSP exceeds voltage threshold, the emited spike will not transmit efficient information to distinct above PSP. Set voltage threshold to maximum activation value can avoid this but suffer a huge latency. This is a trade-off, and in (Sengupta), choose quantile p for different datasets. (Li) propose a Bayesian Optimization to find this p value.

$$r_i^l(T) = \sum_j^{M^{l-1}} W_{ij}^l r_j^{l-1} - \frac{a_{max}^l - V_{th}^l}{T} \tag{11}$$

3.3 Spike inherently Error

This is unavoidable in early timestep, mainly caused by iiregular arrving spike. The phenomenon is that the suddenly coming spike or inactivation neuron wrong spike.



(a) Clipping error (b) Even spikes (c) More spikes (d) Fewer spikes

Figure 3: Spike inherently Error

Larger timestep or spike calibration can relieve this error while cannot eliminate it.

4 Method

4.1 adaptive threshold still holds

Let's take a look at existing methods, no matter weight normalization or threshold balancing, they aim at zipping the gap between ANN and SNN. Though, we should be aware that the real advantage of SNN is its sparse spike which simultaneously low-power and brain-Inspired. Current method, however, set threshold voltage as the same in the same layer and these threshold will remain unchanged despite inference time increasing. It ignores a fact that neurons in different regions of brain represent distinct dynamics and process information differently from other regions. The threshold voltage of neurons is also known to have a broad range rather than a single value. Some neuroscience literature indicates that threshold value is variable in the same neuron and threshold variability is a genuine feature of neurons, which reflects adaptation to the membrane potential at a short

timescale. Thus the voltage threshold should different from neurons and timestep. Here we demonstrate that voltage threshold is a function of timestep t and the transmits the equivalent information with the constant threshold.

obmitting the bias, the equation(5) can be rewrited as the following form

$$V_i^l(t) = V_i^l(t-1) + \sum_j^{M^{l-1}} V_{th,j}^l(t) W_{ij}^l \Theta_{t,j}^{l-1} + b_i^l - V_{th,i}^l(t)\Theta_{t,i}^l \tag{12}$$

firing rate during timestep T is computed as

$$r_i^l(T) = \frac{\sum_{t'=1}^{T} V_{th,i}^l(t')\Theta_{t',i}^l}{T} \tag{13}$$

the firing rate relationship in higher layer, it means equation (9) still satisfy so the conversion error form is the same. But note that the residual neuron membrane potential $V_i(T)$ can be faster adjusted, so the spike information could be more efficiency and thus shorten the conversion latency.

Above equation ensures that all the output neurons are used and adjust the neurons' thresholds to the stimuli for which they become specialized.

4.2 Multi-stage adaptive threshold

In vivo, the spiking threshold displays large variability. This phenomenon has been widely observed in the central nervous system, e.g. visual cortex [1, 2], auditory midbrain [17], hippocampus [9], somatosensory cortex [21]. It has been proposed that threshold variability measured in vivo reflects an adaptation of the spike threshold to the membrane potential. To our best knowledge, threshold varies in conversion is fewly used in [12, 3, 16] whild they only use two-stage or heuristic method and still cannot represent the homoeostasis well.

Inspired by this, we propose a adaptive threshold, which is multi-stage and vaires with inference time. The method can be briefly sumed up as: **varies with firing history and input properties**, Specifically, spike threshold is positively correlated with the average V_i preceding spikes and negatively correlated with the rate of depolarization. Also, it is consistent with some other threshold adaptation models: the threshold increases after each spike and decreases if there is no spike. The relationship between threshold and membrane potential and rate of depolarization is described as

$$V_{th}(t+1) = \tau_{mp} V_{th\_mp}(t) + \tau_{rd} V_{th\_rd}(t+1) \tag{14}$$

Where τ_{mp} and τ_{rd} is the time constant of the dynamic tracking threshold $V_{th\_mp}(t)$ and dynamic evoked threshold $V_{th\_rd}(t+1)$ separately.

dynamic tracking threshold(DTT) DTT is a flection of spiking threshold vaires with firing history. It shows that spike threshold depends on preceding membrane potential and tracking the membrane potential at a short timescale due to inactivation of sodium channel[14, 11, 18] or the activation of potassium channels[10, 6]. in [5], the DTT is a similar first-order kinetic equation, we here use steady-state threshold for fitting our SNNs. we use $V_{m,i}^l(t)$ to denote the average membrane potential during timestep t in layer l neuron i, then DTT is following:

$$V_{th\_mp}^l(t) = \left(\alpha \left(V_i^l(t) - V_m^l(t) \right) + V_T^l + k_a ln \left(1 + e^{\frac{V_i^l(t) - V_m^l(t)}{k_i}} \right) \right) \tag{15}$$

here $\eta, k$is both time constant, V_T^l is the parameters to optimize. when PSP is less than average membrane potential $V_m^l(t)$, the slope is η on the left side of the knee. The slope on the right side is $\frac{k_a}{k_i} + \alpha$. The curvature C is determined by $\alpha, k_a, k_i, V_T^l, V_i^l(t)$.

idealy, when spike reaches to stabality, the $V_i^l(t) \to V_m^l(t)$ so DTT term will be very small. the threshold increas with membrane potential and thus any voltage fluctuations that are slower than threshold adaptation should not have an impact on output spiking, this is indirectly relieve the spike inherently error.

dynamic evoked threshold(DET) DET is a flection of spiking threshold vaires with input properties. paper

$$V_{th\_rd}^{l}(t+1) = \tau_{rd}\left(e^{-|\mu\left(V_i^l(t)\right)|} + e^{-\frac{\left(V_i^l(t+1)-V_i^l(t)\right)}{C}}\right) \tag{16}$$

idealy, when spike reaches to stability, the $V_i^l(t+1) \to V_i^l(t)$ so DET term will be very small.

Interaction of DET and DTT Take together, the causal link between preceding spike membrane potential and neg- atively correlated with the rate of depolarization, shows that threshold adaptation neurons selective to fast input variations and remarkably insensitive to slow ones. In other words, the slow voltage fluctuations are filtered out by threshold adaptation.

Thus our adaptive threshold can be formed as:

$$V_{th,i}^{l}(t+1) = \tau_{mp}\left(\eta\left(V_i^l(t) - V_m^l(t)\right) + ln\left(1 + e^{\frac{V_i^l(t)-V_m^l(t)}{\psi}}\right)\right) + \tau_{rd}\left(e^{-|\mu\left(V_i^l(t)\right)|} + e^{-\frac{\left(V_i^l(t+1)-V_i^l(t)\right)}{C}}\right) \tag{17}$$

Above equation ensures that all the output neurons are used and adjust the neurons thresholds to the stimuli for which they become specialized. The pseudocodes for adaptive threshold algorithm are shown in Algorithm 1.

Algorithm 1 Conversion from ANN to SNN: Multi-stage adaptive threshold(# todo)

Require: Pretrained ANN, training set, SNN's inference timestep T
Ensure: The converted SNN firing rate approximate ANN activation value with shorter latency

for s = 1 to # of samples **do**
  $a_l \leftarrow$ layer-wise activation value
  for l = 1 to L **do**
    $V_{th}^{l} \leftarrow \frac{1}{2} max[V_{th}^{l}, max(a_l)]$
    $SNN.layer[l].V_{th} \leftarrow V_{th}^{l}$
  end for
end for
for t = 1 to timestep T **do**
  for l = 1 to L **do**
    for j = 1 to neuron number of layer l **do**
      $dV_{th} \leftarrow \gamma(SNN.layer[l].R[j] - SNN.layer[l].V_{mem}[j])$
      $SNN.layer[l].V_{th}[j] \leftarrow SNN.layer[l].V_{th}[j] + dV_{th}$
    end for
  end for
end for

5 EXPERIMENTS

Do not change any aspects of the formatting parameters in the style files. In particular, do not modify the width or length of the rectangle the text should fit into, and do not change font sizes (except perhaps in the **References** section; see below). Please note that pages should be numbered.

Table x: Classification accuracy on CIFAR and ImageNet for our converted SNNs, and compared to other conversion methods and ANN.

Table x: detection mAP on VOC and COCO for our converted SNNs, and compared to other conversion methods and ANN.

6 discussion

Indeed a trivial solution to the fitting problem is the threshold model defined by and th 0 ms: the spike threshold always equals the membrane potential, in particular at the upstroke of spikes. To

Table 1: Experimental results on CIFAR100

Method	Use DT	ANN	SNN Best	T=32	T=64	T=128	T=256
VGG16, CIFAR100							
p-Norm [19]	✗	78.49	58.44	44.88	51.89	56.02	58.44
Channel-Norm[13]	✗	78.49	74.74	54.03	67.34	72.50	74.73
Spike-Norm[20]	✗	71.22	70.77	-	-	-	-
TSC[7]	✗	71.22	70.97	-	-	69.86	70.65
RMP-SNN[8]	✗	71.22	70.93	-	-	63.76	68.34
Opt.[4]	✗	77.89	77.71	7.64	21.84	55.04	73.54
Calibration[16]	✗	77.89	77.87	73.55	76.64	77.40	77.68
Burst [15]	✗	78.49	78.71	74.98	78.26	78.66	78.65
This Work	✓	78.49	**78.26**	**34.24**	**63.05**	**75.66**	**78.25**
ResNet20, CIFAR100							
p-Norm [19]	✗	80.69	67.35	38.13	58.09	64.96	67.33
Channel-Norm[13]	✗	80.69	71.26	52.59	66.05	70.08	71.26
Spike-Norm[20]	✗	68.72	64.09	-	-	-	-
TSC[7]	✗	68.72	68.18	-	-	58.42	65.27
RMP-SNN[8]	✗	68.72	67.82	27.64	46.91	57.69	64.06
Opt.[4]	✗	77.16	77.22	51.27	70.12	75.81	77.22
Calibration[16]	✗	77.16	77.73	76.32	77.29	77.73	77.63
Burst [15]	✗	80.69	80.72	76.39	79.83	80.52	80.57
This Work	✓	80.69	**80.58**	**62.96**	**76.41**	**79.52**	**80.44**

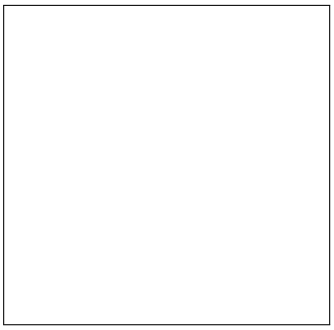


Figure 4: visual Vth varies with inference timestep.

avoid these problems, we instead used the threshold model to predict the occurrence of spikes and their precise timing based only on Vm. The trivial solution mentioned above is a poor predictor of spikes since it would predict too many spikes

Dataset	Method	Network	ANN	SNN Best	$T = 32$	$T = 64$	$T = 128$	$T = 256$
VOC	**Kim's Work (channel-norm)**	Tiny-yolo	-	-	-	-	-	-
	This Work (MSAT)	YOLOv1	-	-	-	-	-	-
COCO	**Kim's Work (channel-norm)**	Tiny-yolo	-	-	-	-	-	-
	This Work (MSAT)	YOLOv1	-	-	-	-	-	-

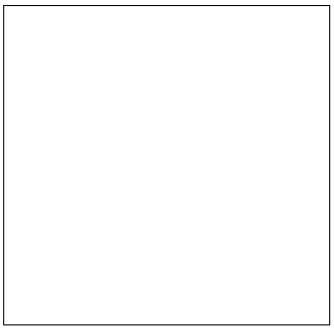


Figure 5: Spike count(efficiency) fig.

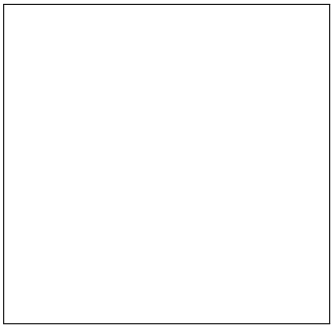


Figure 6: Vth vaires result vs inherent inference timestep.

References

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

[2] Rony Azouz and Charles M Gray. Adaptive coincidence detection and dynamic gain control in visual cortical neurons in vivo. *Neuron*, 37(3):513–523, 2003.

[3] Yunhua Chen, Yingchao Mai, Ren Feng, and Jinsheng Xiao. An adaptive threshold mechanism for accurate and efficient deep spiking convolutional neural networks. *Neurocomputing*, 469:189–197, 2022.

[4] Shikuang Deng and Shi Gu. Optimal conversion of conventional artificial neural networks to spiking neural networks. *arXiv preprint arXiv:2103.00476*, 2021.

[5] Bertrand Fontaine, José Luis Peña, and Romain Brette. Spike-threshold adaptation predicted by membrane potential dynamics in vivo. *PLoS computational biology*, 10(4):e1003560, 2014.

[6] Ethan M Goldberg, Brian D Clark, Edward Zagha, Mark Nahmani, Alev Erisir, and Bernardo Rudy. K+ channels at the axon initial segment dampen near-threshold excitability of neocortical fast-spiking gabaergic interneurons. *Neuron*, 58(3):387–400, 2008.

[7] Bing Han and Kaushik Roy. Deep spiking neural network: Energy efficiency through time based coding. In *European Conference on Computer Vision*, pages 388–404. Springer, 2020.

[8] Bing Han, Gopalakrishnan Srinivasan, and Kaushik Roy. Rmp-snn: Residual membrane potential neuron for enabling deeper high-accuracy and low-latency spiking neural network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13558–13567, 2020.

[9] DA Henze and G Buzsáki. Action potential threshold of hippocampal pyramidal cells in vivo is increased by recent spiking activity. *Neuroscience*, 105(1):121–130, 2001.

[10] Matthew H Higgs and William J Spain. Kv1 channels control spike threshold dynamics and spike timing in cortical pyramidal neurones. *The Journal of physiology*, 589(21):5125–5142, 2011.

[11] Wenqin Hu, Cuiping Tian, Tun Li, Mingpo Yang, Han Hou, and Yousheng Shu. Distinct contributions of nav1. 6 and nav1. 2 in action potential initiation and backpropagation. *Nature neuroscience*, 12(8):996–1002, 2009.

[12] Seijoon Kim, Seongsik Park, Byunggook Na, Jongwan Kim, and Sungroh Yoon. Towards fast and accurate object detection in bio-inspired spiking neural networks through bayesian optimization. *IEEE Access*, 9:2633–2643, 2020.

[13] Seijoon Kim, Seongsik Park, Byunggook Na, and Sungroh Yoon. Spiking-yolo: spiking neural network for energy-efficient object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11270–11277, 2020.

[14] Hiroshi Kuba, Yuki Oichi, and Harunori Ohmori. Presynaptic activity regulates na+ channel distribution at the axon initial segment. *Nature*, 465(7301):1075–1078, 2010.

[15] Yang Li and Yi Zeng. Efficient and accurate conversion of spiking neural network with burst spikes. *arXiv preprint arXiv:2204.13271*, 2022.

[16] Yuhang Li, Shikuang Deng, Xin Dong, Ruihao Gong, and Shi Gu. A free lunch from ann: Towards efficient, accurate spiking neural networks calibration. In *International Conference on Machine Learning*, pages 6316–6325. PMLR, 2021.

[17] Jose Luis Pena and Masakazu Konishi. From postsynaptic potentials to spikes in the genesis of auditory spatial receptive fields. *Journal of Neuroscience*, 22(13):5652–5658, 2002.

[18] Jonathan Platkiewicz and Romain Brette. Impact of fast sodium channel inactivation on spike threshold dynamics and synaptic integration. *PLoS computational biology*, 7(5):e1001129, 2011.

[19] Bodo Rueckauer, Iulia-Alexandra Lungu, Yuhuang Hu, Michael Pfeiffer, and Shih-Chii Liu. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in neuroscience*, 11:682, 2017.

[20] Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy. Going deeper in spiking neural networks: Vgg and residual architectures. *Frontiers in neuroscience*, 13:95, 2019.

[21] W Bryan Wilent and Diego Contreras. Stimulus-dependent changes in spike threshold enhance feature selectivity in rat barrel cortex neurons. *Journal of Neuroscience*, 25(11):2983–2991, 2005.

A Appendix

Optionally include extra information (complete proofs, additional experiments and plots) in the appendix. This section will often be part of the supplemental material.