ICML | 2022

AutoSNN:
Towards Energy-Efficient Spiking Neural Networks

Byunggook Na

Jisoo Mok

Seongsik Park

Dongjin Lee

Hyeokjun Choe

Sungroh Yoon

SEOUL
NATIONAL
UNIVERSITY

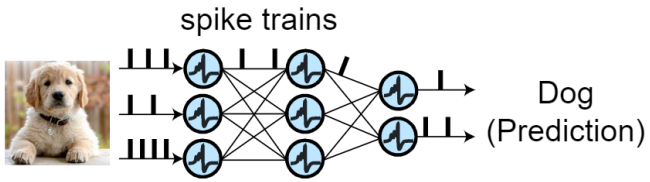
SAMSUNG

Samsung Advanced
Institute of Technology

KIST
Korea Institute of
Science and Technology

Spiking Neural Networks (SNNs)

- Information transmission through spikes instead of activation values

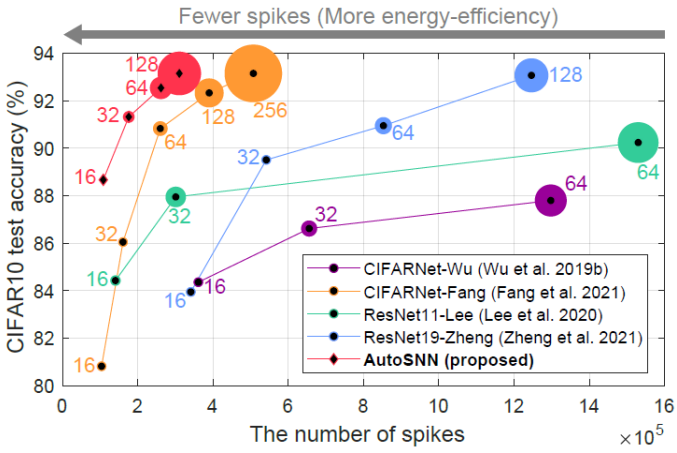


- Energy efficiency
  - # of spikes = energy consumption
  - Sparse spiking events and event-driven computation
  - HW support: neuromorphic chips (ex. IBM's TrueNorth, Intel's Loihi)

Motivation and Overview

- For SNNs, the suitability of architectures has been rarely investigated.

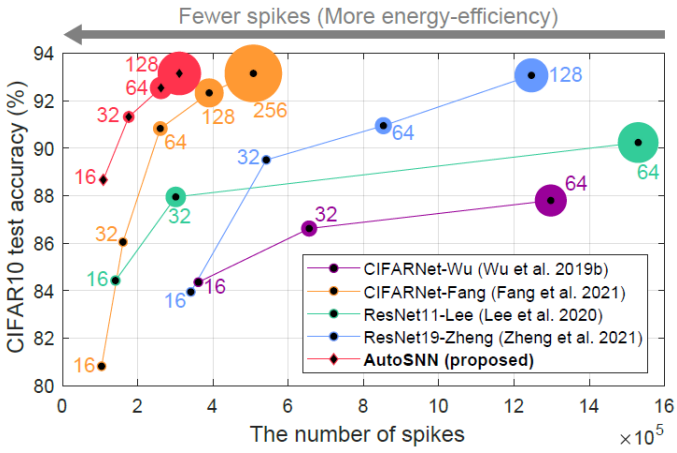
- We leverage neural architecture search (NAS) to find more suitable SNN architectures.



Motivation and Overview

- For SNNs, the suitability of architectures has been rarely investigated.

- We leverage neural architecture search (NAS) to find more suitable SNN architectures.



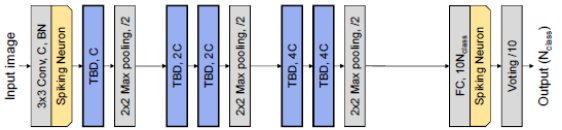
Q. How do we design NAS pipeline to search for energy-efficient SNNs?

➔ Contribution 1. Analyze and propose desirable design choice of SNN search space

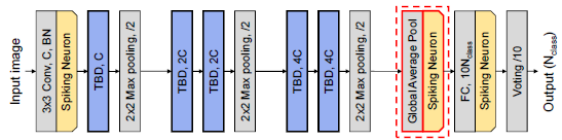
➔ Contribution 2. Propose spike-aware evolutionary search algorithm

Architectural Analysis and Search Space Design

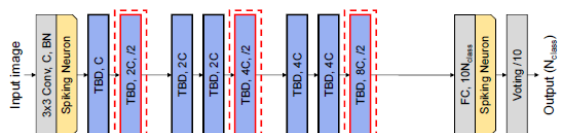
Base architectures



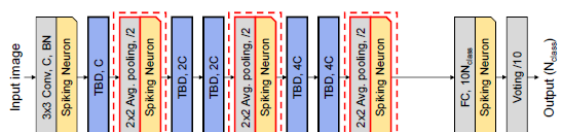
(a) SNN_1 (proposed macro architecture for AutoSNN)



(b) SNN_2 (SNN_1 with GAP)

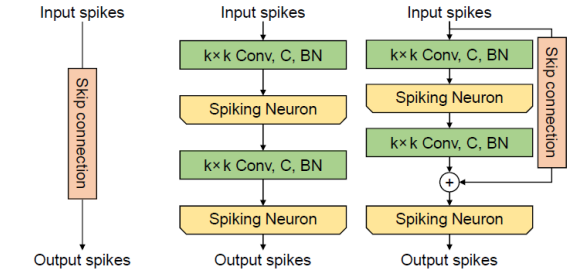


(c) SNN_3 (TBD blocks for down-sampling in SNN_1)



(d) SNN_4 (Average pooling layers for down-sampling in SNN_1)

Spiking building blocks



(a) Skip connection (Skip) (b) Spiking convolution block (SCB) (c) Spiking residual block (SRB)

Spiking inverted bottleneck block (SIB)

Architectural Analysis and Search Space Design

Architecture	GAP	Normal	Down-sample	Acc.(%)	Spikes
SNN_1	✘	SCB_k3	MaxPool	86.93	154K
SNN_2	✔	SCB_k3	MaxPool	85.05	168K

Normal	Down-sample	Acc.(%)	Spikes
SRB_k3	MaxPool	87.54	146K
SRB_k3	MaxPool	85.82	168K

1) A global average pooling (GAP) layer with spiking neurons is not suitable for SNNs.

Architectural Analysis and Search Space Design

Architecture	GAP	Normal	Down-sample	Acc.(%)	Spikes
SNN_1	✘	SCB_k3	MaxPool	86.93	154K
SNN_2	✔	SCB_k3	MaxPool	85.05	168K
SNN_3	✘	SCB_k3	SCB_k3	87.94	222K
SNN_4	✘	SCB_k3	AvgPool	79.59	293K

Normal	Down-sample	Acc.(%)	Spikes
SRB_k3	MaxPool	87.54	146K
SRB_k3	MaxPool	85.82	168K
SRB_k3	SRB_k3	89.18	221K
SRB_k3	AvgPool	83.79	291K

1) A global average pooling (GAP) layer with spiking neurons is not suitable for SNNs.

2) Max pooling layers are best-suited for down-sampling in SNNs.

Architectural Analysis and Search Space Design

Architecture	GAP	Normal	Down-sample	Acc.(%)	Spikes
SNN_1	✘	SCB_k3	MaxPool	86.93	154K
SNN_2	✔	SCB_k3	MaxPool	85.05	168K
SNN_3	✘	SCB_k3	SCB_k3	87.94	222K
SNN_4	✘	SCB_k3	AvgPool	79.59	293K

Normal	Down-sample	Acc.(%)	Spikes
SRB_k3	MaxPool	87.54	146K
SRB_k3	MaxPool	85.82	168K
SRB_k3	SRB_k3	89.18	221K
SRB_k3	AvgPool	83.79	291K

1) A global average pooling (GAP) layer with spiking neurons is not suitable for SNNs.

2) Max pooling layers are best-suited for down-sampling in SNNs.

Spiking block in SNN_1	Acc. (%)	Spikes	Firing rates
SCB_k3	86.93	154K	0.18
SRB_k3	87.54	146K	0.17
SIB_k3_e1	81.07	243K	0.23
SIB_k3_e3	88.45	374K	0.17

Spiking block	$\lambda_{reg} = 1$ Acc.	Spikes	$\lambda_{reg} = 0.1$ Acc.	Spikes	$\lambda_{reg} = 0.01$ Acc.	Spikes
SCB_k3	64.36	83K	79.09	84K	86.39	124K
SRB_k3	72.76	49K	83.25	70K	86.59	109K
SIB_k3_e1	56.61	89K	73.54	119K	81.05	155K
SIB_k3_e3	74.71	136K	84.59	186K	87.61	249K

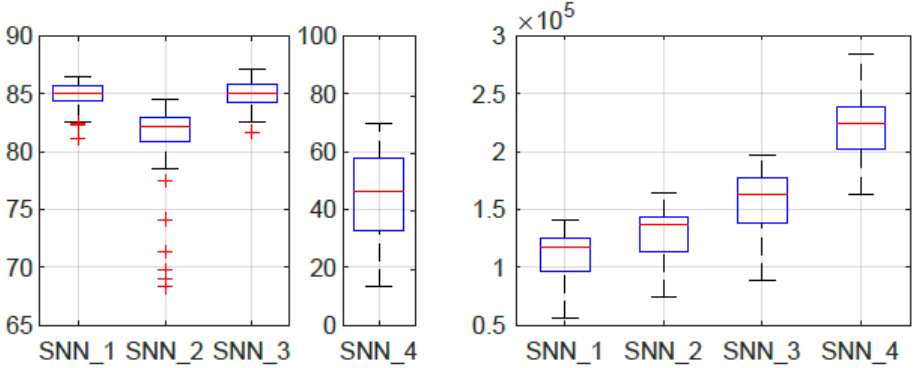
※ with spike regularization

3) Spiking inverted bottleneck blocks generate large number of spikes.

Search Space Quality

- Based our findings, we defined our search space based on SNN_1

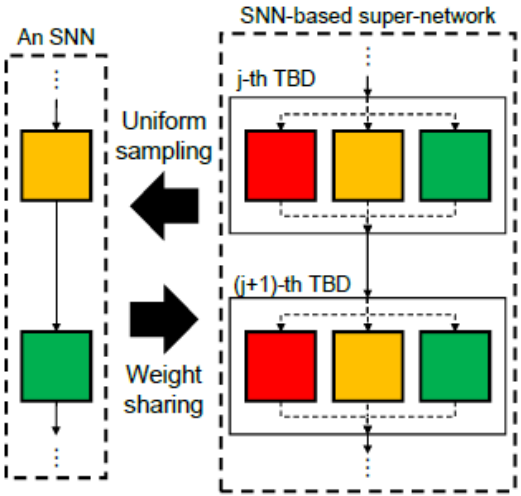
- When training 100 architectures that are randomly sampled from each search space:



(a) CIFAR10 test accuracy (%) **(b) The number of spikes**

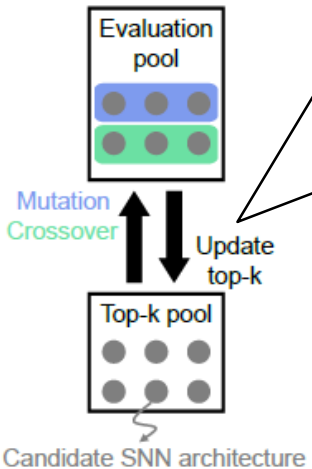
One-shot NAS and Spike-aware Evolutionary Search

Using training data



(a) Train an SNN-based super-network

Using validation data



(b) Evolutionary search

Spike-aware fitness

$$F(A) = Accuracy \times (N/N_{\text{avg}})^{\lambda}$$

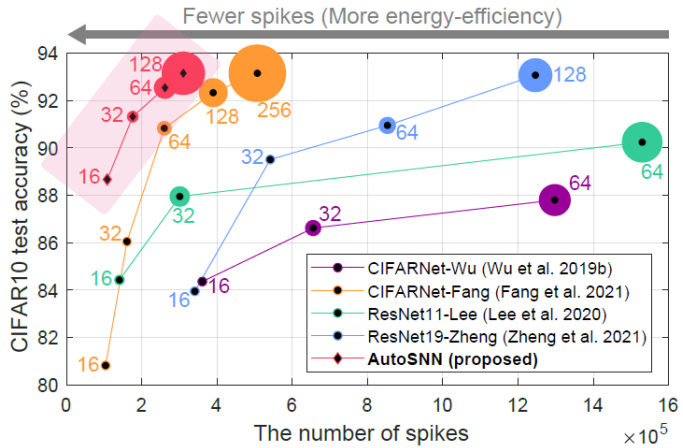
N: # of spikes of architecture A w.r.t. validation data

N_{avg}: # of averaged spikes of all architectures w.r.t training data

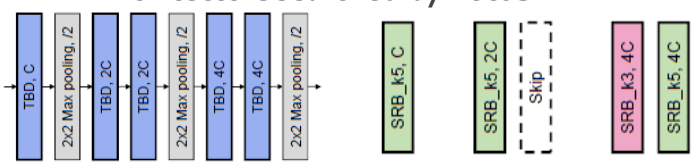
$\lambda < 0$: to minimize # of spikes

Results

Frontier performance in the acc-spikes region



Architecture searched by AutoSNN



Transferability to other datasets (static datasets, neuromorphic datasets)

Data	SNN Architecture	C	Acc (%) ↑	Spikes ↓
CIFAR100	Fang et al. 2021	256	66.83	716K
	AutoSNN	64	69.16	326K
SVHN	Fang et al. 2021	256	91.38	462K
	AutoSNN	64	91.74	215K
Tiny-Image Net-200	Fang et al. 2021	256	45.43	1724K
	AutoSNN†	64	46.79	680K
CIFAR10 -DVS	Wu et al. 2019b	128	‡60.50	-
	Fang et al. 2021	128	69.10	4521K
	Zheng et al. 2021	64	66.10	1550K
	AutoSNN†	16	72.50	1269K
DVS128 -Gesture	He et al. 2020	64	‡93.40	-
	Kaiser et al. 2020	64	‡95.54	-
	Fang et al. 2021	128	95.49	1459K
	Zheng et al. 2021	64	96.53	1667K
	AutoSNN†	16	96.53	423K

Summary

- Goal: **How to facilitate NAS** for SNN domain that overlooks architectural importance

- **AutoSNN**
  - **Architectural analysis on accuracy and energy-efficiency**
    - Derive the energy-efficient search space
  - **Spike-aware evolutionary search**
    - Super-network training: direct training method for SNNs
    - Searching: evolutionary search algorithm with spike-aware fitness design
  - Demonstrate the superiority of AutoSNN on various datasets

paper

code

Contact: byunggook.na@gmail.com, sryoon@snu.ac.kr