Temporal Pulses Driven Spiking Neural Network for Time and Power Efficient Object Recognition in Autonomous Driving

Wei Wang[1], Shibo Zhou[2], Jingxi Li[2], Xiaohua Li[2], Junsong Yuan[1], Zhanpeng Jin[1]

[1] University at Buffalo, [2] Binghamton University

University at Buffalo
Department of Computer Science and Engineering
School of Engineering and Applied Sciences

Binghamton University
State University of New York

OUTLINE

Introduction

- Objective
  - Explore a new paradigm for object recognition in autonomous driving:
    - Excellent accuracy
    - Ultra low power consumption and time delay
- Contribution
  - An event driven model

    Spiking neural networks that can directly process raw LiDAR pulses (without point-cloud or voxelization).
  - *Sim LiDAR* dataset

    A comprehensive temporal pulses dataset that simulates LiDAR reflection of different road conditions and target objects in diverse noise environments.
  - Extraordinary time- and energy-efficiency on real-world data

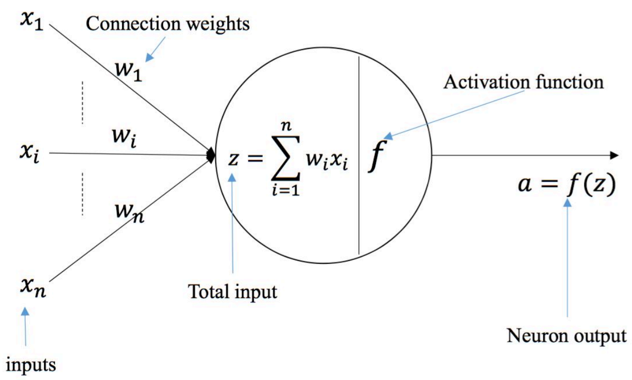
Motivation

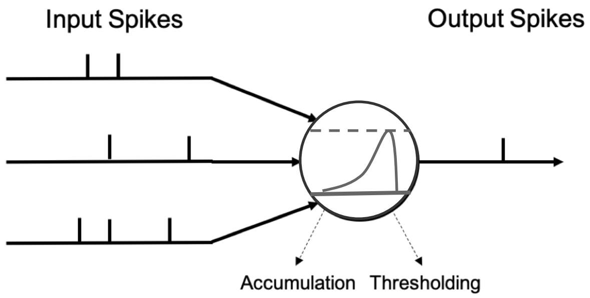
- Why Spiking Neural Network (SNN) (rather than conventional neural network)
  - **Data representation:** Spikes have inherent temporal information.
  - **Info. processing:** Event driven, asynchronism
  - **Hardware friendly:** energy efficient

- Why Raw Temporal Pulses (rather than point-clouds/voxels)
  - Eliminates the restrictions of frames.
  - Can achieve better time efficiency
  - Have less computational overhead.

Rationale & Methods

❑ Neuronal Model





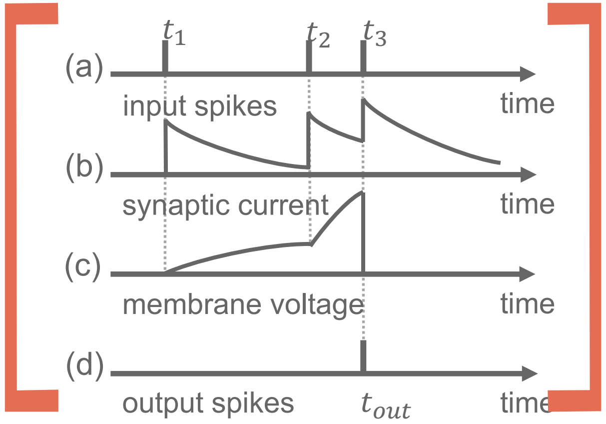
	ANN	SNN
Input / Output	continuous real value / non-sparse	discrete spikes / sparse
Neuronal Model	linear combination + activation (weighted sum)	accumulation + thresholding (integrate-and-fire)

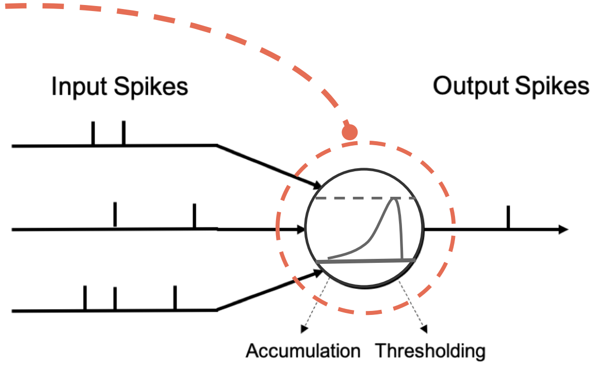
Biological Plausible

Rationale & Methods

❑ Neuronal Model

- non-leaky integrate and fire (n-LIF) neuron [1]





Integrate:

- Synaptic current *I*: when there is a spike propagating
- Membrane voltage *V*: accumulates from the current

Fire:

- generate a spike when *V* > threshold

$$\frac{dv_j(t)}{dt} = \sum_i w_{ji}\kappa(t - t_i)$$

$$\kappa(t) = u(t)e^{-\frac{t}{\tau}} \qquad u(t) = \begin{cases} 1, & \text{if } t \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

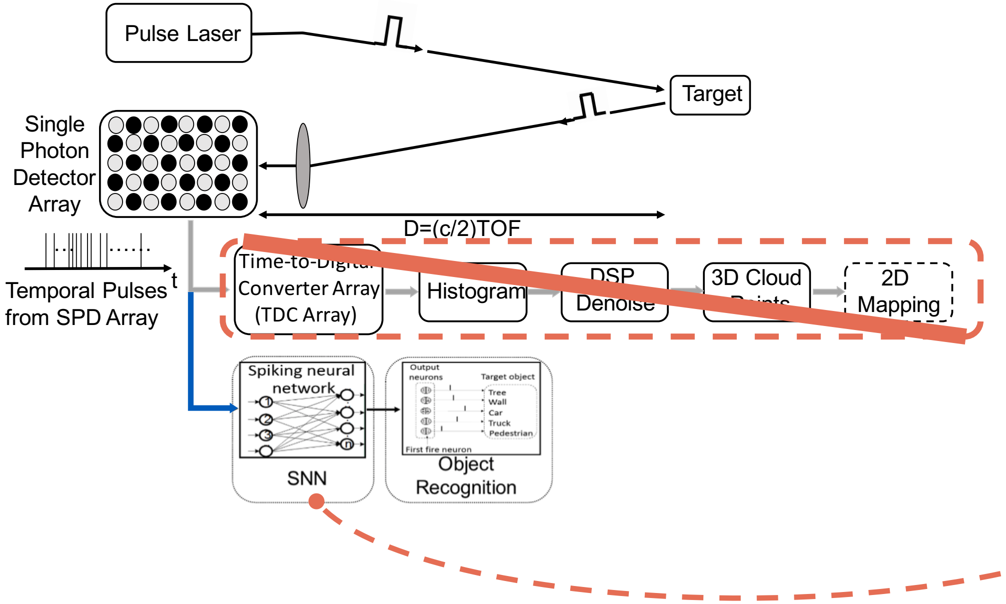
[1] H. Mostafa, "Supervised learning based on temporal coding in spiking neural networks", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 7, pp. 3227–3235, 2018.

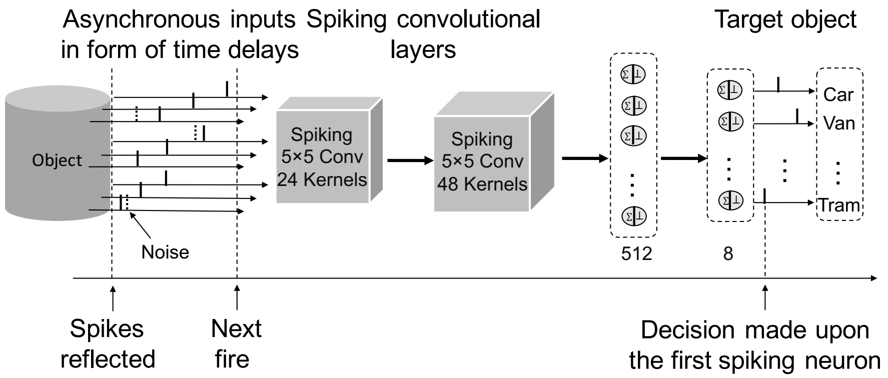
Rationale & Methods

- Encoding Techniques of Spike Train
  - Spike count / rate ✗
  - Temporal coding (arrival/interval time) ✓

$$e^{t_j} = \sum_{i \in C} e^{t_i} \frac{w_{ji}}{\sum_{\ell \in C} w_{j\ell} - 1}$$

- System Flow & Network structure





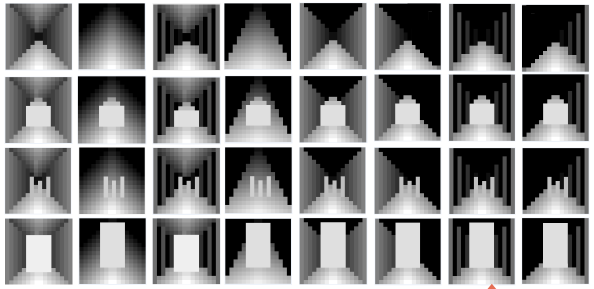
Experiment & Evaluation

Datasets

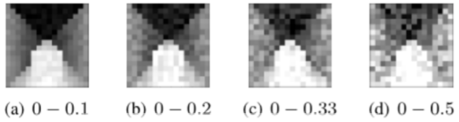
- Sim LiDAR
  - Arrival time
  - 32 categories
  - Noise injection
- KITTI
  - Truncated
  - 8 categories
  - 32,456/8,000 samples (train/test)
- Dynamic visual sensor (DVS) dataset [2]
  - event-based camera (asynchronous)
  - 36 categories (a ~ z, 0 ~ 9)
  - 3,453/3,000 samples (train/test)

Objects:
car, pedestrian, truck

Road conditions:
tunnel, open road
lower/upper bridge
road (walls on one/two side)
road (lamps on one/two side)



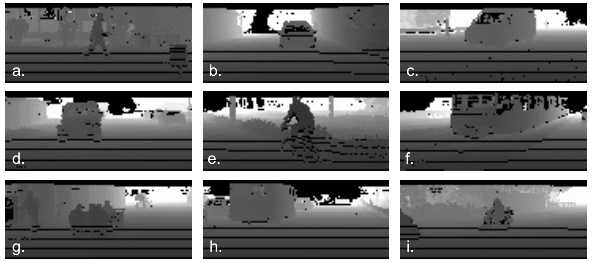
Sim LiDAR covers different objects and road conditions

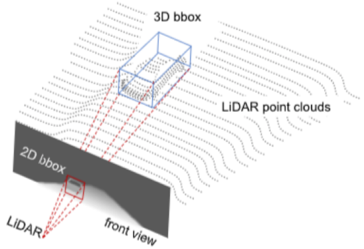


Diverse noise levels

Converted to gray-scale images ONLY for visualization

Objects:
Pedestrian (a)
Car (b), Van (c)
Truck (d), Cyclist (e)
Tram (f)
Person sitting (g)
Misc (h, i)





Truncation of KITTI with examples

DVS barrel dataset [2]

[2] G. Orchard, C. Meyer, R. Etienne-Cummings, C. Posch, N. Thakor, and R. Benosman, “HFirst: a temporal approach to object recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37,no. 10, pp. 2028–2040, 2015.

Experiment & Evaluation

Results

- **DVS Dataset**

  (compare with existing models)

Model	Method	Accuracy
[2]	HFirst Temporal	84.9%
[3]	CNN Spike-based	91.6%
[3]	CNN Frame-based	95.2%
Our model	Spiking MLP	**99.5%**

- **KITTI**

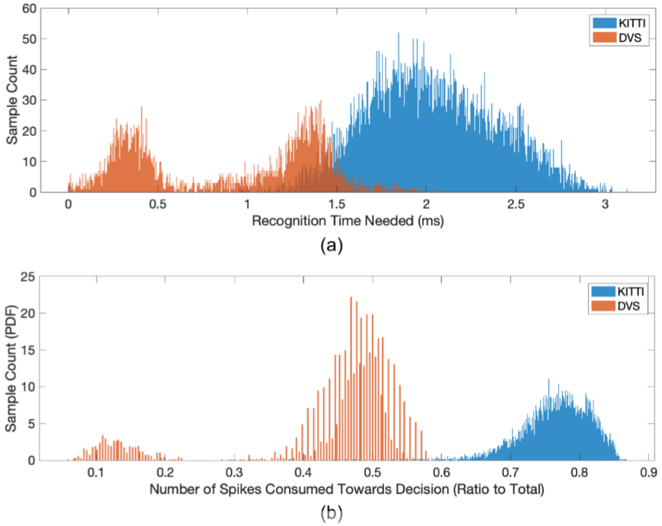
  (compare with conventional CNN)

Model	Acc. (%)	T_{rec} (ms)	R_{data} (%)	Power Consumption (α=0.37 pJ / α=45 pJ)
SCNN	96.62	2.02	76	29.83 nJ/3.63 μJ
CNN	88.22	2.58	100	0.67 J
VGG-16	92.72	11.34	100	2.95 J
ResNet-50	92.84	71.30	100	18.54 J

- **Sim LiDAR**

  (robustness against noise)

Noise Range	0 - 0.10	0 - 0.20	0 - 0.33	0 - 0.50
Accuracy	99.83%	96.16%	82.66%	68.16%



Efficiency of proposed model

data ratio:

$$R_{data} = \frac{N_{\text{contributing\_pulses}}}{N_{\text{all\_pulses}}}$$

The proposed SNN only needs part of the input pulses!

[2] G. Orchard, C. Meyer, R. Etienne-Cummings, C. Posch, N. Thakor, and R. Benosman, “HFirst: a temporal approach to object recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37,no. 10, pp. 2028–2040, 2015.

[3] J. A. P´erez-Carrasco, B. Zhao, C. Serrano, B. Acha, T. Serrano-Gotarredona, S. Chen, and B. Linares-Barranco, “Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing–application to feedforward ConvNets”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 11, pp. 2706–2719, 2013

Discussion & Conclusion

Conclusion

- The proposed SNN has remarkable accuracy
- Extraordinary time and energy efficiency
- Great potential in resource- and/or time-constrained applications

Future Work

- Event-driven effective and efficient 3D object detection
- Combination with neuromorphic hardware
- End to end event driven solution
  - Event-driven sensor
  - Spiking hardware with LIF unit
  - SNN model

Thank You !