Software Impacts 10 (2021) 100131

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

ConvSNN: A surrogate gradient spiking neural framework for radar gesture recognition

Ali Safa *, Francky Catthoor, Georges G.E. Gielen

imec, Leuven, Belgium
KU Leuven, Leuven, Belgium

ARTICLE INFO

Keywords:
Spiking neural networks
Radar-based gesture recognition
Neuromorphic computing

ABSTRACT

Spiking neural networks (SNNs) have recently gained large interest for edge-AI applications due to their low latency and ultra-low energy consumption. Unlike DNNs, SNNs communicate information using spike trains. As the derivative of spike trains are highly ill-defined, the use of surrogate gradients has been proposed as an efficient method for training SNNs. Still, the lack of open-source SNN softwares and the limited range of demonstrated SNN applications slows down a wider SNN adoption. We release our ConvSNN framework, demonstrating the novel applicability of quantized-weight SNNs for radar gesture recognition. Our framework will facilitate future research in the SNN area.

Code metadata

Current code version	1.0
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2021-111
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/6192245/tree/v1
Legal Code License	MIT License
Code versioning system used	git
Software code languages, tools, and services used	Python
Compilation requirements, operating environments & dependencies	pytorch v.1.5.1, sklearn, matplotlib, numpy
If available Link to developer documentation/manual	
Support email for questions	Ali.Safa@imec.be

1. Introduction

In recent years, spiking neural networks (SNNs) have emerged as a new *event-based* computing paradigm (as opposed to classical frame-based networks). SNNs have gained significant interest due to their low inference latency, low energy consumption and their compatibility with the growing number of massively parallel *neuromorphic* computing architectures (investigated by companies such as Intel [1]), making them attractive choices for edge-AI applications [2]. In contrast to the *continuous* activation functions (e.g., ReLU) used in classical deep neural networks (DNNs), SNNs make use of discontinuous, *spiking* activation function, which code information as spike trains (Dirac combs). This leads to ill-defined gradients throughout the network, prohibiting the direct use of *error back-propagation* (backprop). To circumvent this problem and enable backprop, the use of *surrogate gradients* [3] with *back-propagation through time* (BPTT) [4] has been recently proposed, and quickly gained popularity due to its remarkable efficiency [5].

Still, a greater adoption of SNNs has been slowed down, mainly due to (1) the small number of open-source SNN frameworks, and (2) the limited number of demonstrated applications. In this paper, we address both problems by releasing our convolutional SNN (ConvSNN) framework, targeting the novel use-case of SNN-based radar gesture recognition. In addition, our framework also differs from previous SNN works due to its use of quantization-aware training. Our release software demonstrate the applicability of a resource-constrained ConvSNN with 4-bit weights (typical bit width in neuromorphic processors [7]) on two different radar gesture recognition dataset [6,8], achieving more than 91% of accuracy. The proposed system is therefore

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author at: KU Leuven, Leuven, Belgium.
E-mail address: Ali.Safa@imec.be (A. Safa).

https://doi.org/10.1016/j.simpa.2021.100131
Received 31 August 2021; Accepted 3 September 2021

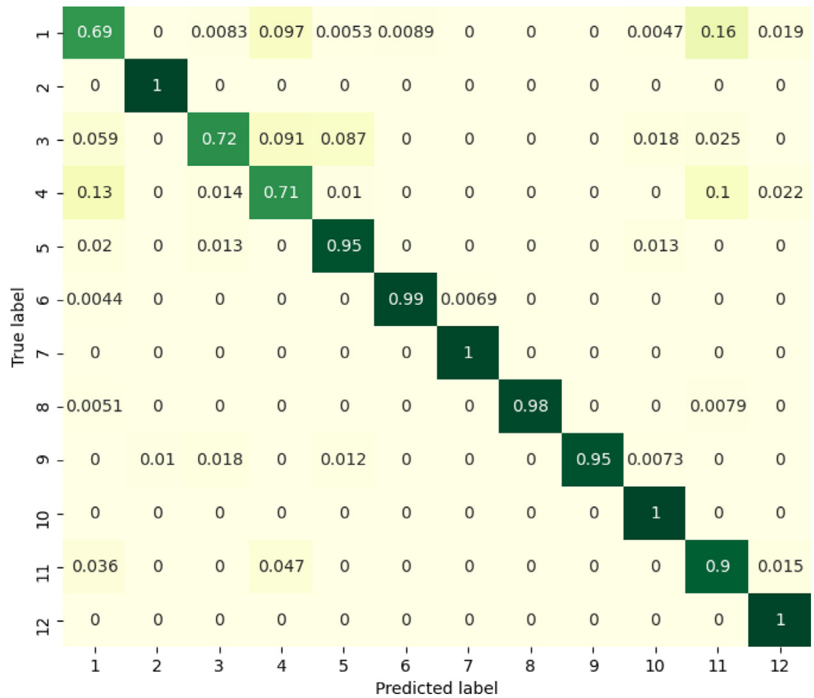

Fig. 1. Confusion matrix for the 12-class radar gesture recognition of [6]. Generated by running the file *SNN_12_class_test.py*. The recognition accuracy is $91\% \pm 2\%$.

ready for implementation in the growing number of ultra-low-power neuromorphic processors [1].

2. Description

The ConvSNN framework is written in Python and relies on the PyTorch library [9] for automatic differentiation and for the definition of a custom spiking neuron class called *ActFun* in the file *eNetworks.py*. The *ActFun* class behaves as a leaky integrate and fire (LIF) neuron in the forward pass and uses a Gaussian surrogate in the backprop pass [3]. Users can easily experiment with different surrogate models by writing their custom mathematical model in the *backward* method of the *ActFun* class. The LIF decay parameter *decay_neu* and the Gaussian surrogate parameters can be modified in the top lines of *eNetworks.py*. The ConvSNN is defined by the class *mini_eCNN* in *eNetworks.py* and can be easily modified by users (e.g., to experiment with more layers). The file *SNN_12_class_train.py* trains the ConvSNN on the 12-class radar dataset. The function *low_precision* is used to quantize the network weights to a settable bit width. The same description holds for the file *SNN_5_class_train.py*, which trains the network on the second radar dataset (5-class). The files *SNN_12_class_test.py* and *SNN_5_class_test.py* test the accuracy of the saved models on their corresponding datasets. Fig. 1 shows the confusion matrix obtained by running *SNN_12_class_test.py*. Models saved during training are stored in the folders *saved_models_12_class* and *saved_models_5_class* and are loaded from there by the test files *SNN_12_class_test.py* and *SNN_5_class_test.py*. The datasets are stored in the folders *dataset_5_class* and *dataset_12_class*.

3. Impact

Our ConvSNN framework has enabled the development of a novel radar gesture recognition system with implementation-ready 4-bit weights [10], targeting the ultra-low-power edge-AI and IoT domains. The framework proposed in this paper is modular and can be easily modified by other researchers to suit their custom needs (as it mostly relies on standard PyTorch functions [9]). Our ConvSNN framework is one of the few state-of-the-art SNN softwares publicly released for the greater machine learning community, and will help researchers to quickly evaluate and extend SNNs, enabling a faster adoption of the emerging *neuromorphic computing* paradigm.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors thank André Bourdoux, Ilja Ocket, Federico Corradi and Lars Keuninckx for the discussions and guidance, and the Flanders AI research program for partially supporting this work.

References

[1] M. Davies, et al., Loihi: A neuromorphic manycore processor with on-chip learning, IEEE Micro 38 (1) (2018) 82–99, http://dx.doi.org/10.1109/MM.2018.112130359.

[2] S. Moradi, N. Qiao, F. Stefanini, G. Indiveri, A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (DYNAPs), IEEE Trans. Biomed. Circuits Syst. 12 (1) (2018) 106–122, http://dx.doi.org/10.1109/TBCAS.2017.2759700.

[3] E.O. Neftci, H. Mostafa, F. Zenke, Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks, IEEE Signal Process. Mag. 36 (6) (2019) 51–63, http://dx.doi.org/10.1109/MSP.2019.2931595.

[4] P.J. Werbos, Backpropagation through time: what it does and how to do it, Proc. IEEE 78 (10) (1990) 1550–1560, http://dx.doi.org/10.1109/5.58337.

[5] Shrestha S.B., Orchard, G, SLAYER: Spike layer error reassignment in time, in: Neural Information Processing Systems, Montreal, Canada, 2018.

[6] S. Wang, J. Song, J. Lien, I. Poupyrev, O. Hilliges, Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum, 2016, pp. 851–860.

[7] C. Frenkel, M. Lefebvre, J. Legat, D. Bol, A 0.086-mm^2 12.7-pJ/SOP 64k-Synapse 256-neuron online-learning digital spiking neuromorphic processor in 28-nm CMOS, in: IEEE Trans. Biomed. Circuit. Syst., 13, (1) 2019, pp. 145–158, http://dx.doi.org/10.1109/TBCAS.2018.2880425.

[8] J. Stuijt, M. Sifalakis, A. Yousefzadeh, F. Corradi, μBrain: An event-driven and fully synthesizable architecture for spiking neural networks, Front. Neurosci. 15 (538) (2021).

[9] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, 2017.

[10] Ali Safa, André Bourdoux, Ilja Ocket, Francky Catthoor, Georges G.E. Gielen, A 2-μJ, 12-class, 91% accuracy spiking neural network approach for radar gesture recognition, 2021.