On the Use of Spiking Neural Networks for Ultralow-Power Radar Gesture Recognition

Ali Safa, *Graduate Student Member, IEEE*, André Bourdoux, *Senior Member, IEEE*, Ilja Ocket, *Member, IEEE*, Francky Catthoor, *Fellow, IEEE*, and Georges G. E. Gielen, *Fellow, IEEE*

***Abstract*— Radar processing via spiking neural networks (SNNs) has recently emerged as a solution in the field of ultralow-power wireless human–computer interaction. Compared to traditional energy- and area-hungry deep learning methods, SNNs are significantly more energy-efficient and can be deployed in the growing number of compact SNN accelerator chips, making them a better solution for ubiquitous IoT applications. We propose a novel SNN strategy for radar gesture recognition, achieving more than 91% of accuracy on two different radar datasets. Our work significantly differs from previous approaches as: 1) we use a novel radar-SNN training strategy; 2) we use quantized weights, enabling power-efficient implementation in real-world SNN hardware; and 3) we report the SNN energy consumption per classification, clearly demonstrating the real-world feasibility and power savings induced by SNN-based radar processing. We release an evaluation code to help future research.**

***Index Terms*— Radar gesture recognition, spiking networks.**

I. Introduction

WIRELESS human–computer interaction using radar-based gesture recognition systems has attracted large interest during the past decade, enabling applications such as smart domotics, AR/VR headsets, and many other *touchless* interfacing solutions that are key for a more hygienic, post-COVID-19 world [1]. In order to embed radar sensing into ubiquitous, ultralow-power IoT devices, research at the hardware side has mainly been devoted to high-level integration of radar transceivers [2] with a focus on energy and area efficiency [3]. In contrast, research at the signal processing side has mainly been devoted to the use of high-accuracy deep neural networks (DNNs) known to be rather energy- and area-hungry [4], [6]. State-of-the-art DNN-based techniques that achieve high-class (>10), high-performance (>90%) gesture recognition either rely on the use of an expensive desktop-grade GPU [4] or either on the use of a lower power embedded GPU [6]. A novel DNN targeting mobile phones has been proposed in [5], trading a lower number of classes for lower memory consumption, still unsuited for ultralow-power IoT.

Manuscript received September 29, 2021; revised October 30, 2021; accepted November 3, 2021. This work was supported by the Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen Programme. *(Corresponding author: Ali Safa.)*

Ali Safa, Ilja Ocket, Francky Catthoor, and Georges G. E. Gielen are with imec, 3001 Leuven, Belgium, and also with the Department of Electrical Engineering, KU Leuven, 3001 Leuven, Belgium (e-mail: ali.safa@imec.be; ilja.ocket@imec.be; francky.catthoor@imec.be; georges.gielen@kuleuven.be).

André Bourdoux is with imec, 3001 Leuven, Belgium (e-mail: andre.bourdoux@imec.be).

Data is available on-line at https://tinyurl.com/yu598c7e.

Color versions of one or more figures in this letter are available at https://doi.org/10.1109/LMWC.2021.3125959.

Digital Object Identifier 10.1109/LMWC.2021.3125959

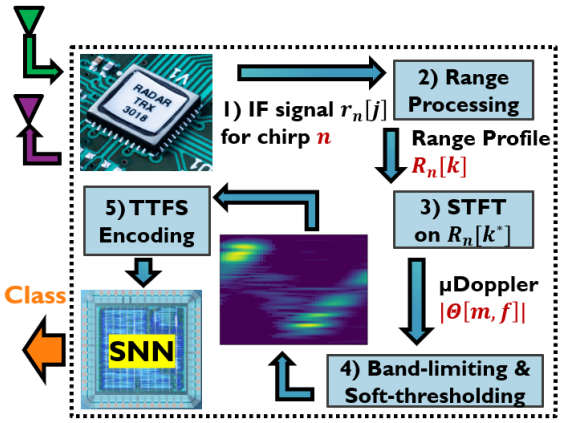

Fig. 1. μDoppler-based radar-SNN architecture proposed to solve the five-class 8-GHz dataset of [7] with 93% of accuracy.

Very recently, the use of energy-efficient spiking neural networks (SNNs) for radar processing has grown to become an emerging topic in radar sensing and is currently being investigated by many teams [7]–[10]. Algorithm-wise, SNNs differ from DNNs as they communicate interneural information asynchronously using binary spikes that are only emitted when the neuron membrane potential reaches a specific threshold. In contrast to DNNs, SNNs do not require expensive multiply–accumulate operations at the input of each neuron but make use of inexpensive add operations only. Hardware-wise, SNNs can be integrated near the radar sensor (see Fig. 1) as subthreshold analog circuits, reaching more than five orders of magnitude lower power consumption compared to embedded GPUs [6], [12], [13].

Still, the development of SNN-based radar processing is at an early stage. In this letter, our aim is to propose a novel SNN architecture for radar gesture recognition using a different, surrogate gradient-based approach than the ones used in previously presented radar-SNN systems [20]. Compared to previous works [7], [9], [10], which either use the μDoppler preprocessing [14] or the range-Doppler preprocessing [4], we demonstrate that our novel radar-SNN approach is compatible with both preprocessing techniques. In contrast to the work in [8], our approach is purely SNN-based, while the system of [8] uses an SNN followed by classical machine learning techniques such as random forest, which cannot be deployed in subthreshold analog SNN circuits. Compared to [8]–[10], our system uses implementation-ready, quantized weights (typical bit width in SNN hardware is <8 bits [7], [11]), while

none of the aforementioned works quantize their weights, making their reported performances (85%–98%) unclear when deployed in real-world hardware (12-class 91% with 6-bit weights and 5-class 93% with 4-bit weights in our work). Finally, in contrast to most previous works [8]–[10], we report an estimate of our SNN energy consumption when deployed in SNN hardware [7]. We assess our system on two radar datasets: the 12-class Google Soli dataset of [4] and the 5-class 8-GHz dataset of [7] to enable comparison with the radar-SNN system in [7].

II. Radar-SNN Processing Pipeline

A. 5-Class 8-GHz Dataset and Preprocessing

The dataset of [7] contains radar ADC data with $N_{\text{chirps}} = 192$ chirps per frame and with a variable number of frames per gesture acquisition N_{frames} (step 1 in Fig. 1). μDoppler signatures [14] are acquired for each gesture acquisition in the dataset by first computing the range profiles $R_n[k]$ for each chirp $n = 1, \ldots, N_{\text{tot}}$ (where N_{tot} is the total number of chirps). $R_n[k]$ is acquired by DFT using a Blackman window [15] (step 2 in Fig. 1). Then, we apply the short-time Fourier transform (STFT) to the sequence $\tilde{R}_n[k^*] = R_n[k^*] - R_{n-1}[k^*]$ (step 3 in Fig. 1), which removes the strong dc component during each analysis window [16] as follows:

$$\Theta[m, f] = \sum_{n=-\infty}^{\infty} \tilde{R}_n[k^*] g_s[n - mR] e^{-j2\pi f n} \tag{1}$$

where k^* denotes the range bin where the gestures are executed, g_s denotes a *Hanning* window of length s, and R is the hop size ($s = 192$ and $R = 8$ throughout this letter). k^* is known *a priori* as the gestures are executed at 2 m from the radar. We define the μDoppler signature as $|\Theta[m, f]|$, which is a matrix of size $(N_T \times s)$ with N_T given by [10]

$$N_T = \left\lfloor \frac{N_{\text{frames}} N_{\text{chirps}} - N_{\text{overlap}}}{R} \right\rfloor \tag{2}$$

where $N_{\text{overlap}} = s - R$ is the number of overlapping bins between successive windows. Radar maps to be fed to the SNN are obtained by cutting $|\Theta[m, f]|$ along dimension m into images of 48 time samples. $\lfloor (N_T/48) \rfloor$ examples are thus obtained for each acquisition. By balancing the dataset and by removing the first and the last six example maps to remove startup (when the human simply sits in front of the radar before performing gestures) and ending artifacts (when the human reaches out to the radar to stop it), we obtain a balanced dataset with a total of 1695 μDoppler examples.

Each example map is then normalized between $[0, 1]$. Out-of-band noise is removed through the band-limiting of the Doppler frequency axis by keeping the normalized frequency range between $[-0.26, 0.26]$ only. This frequency band was identified visually by evaluating the maximal significant extent of the Doppler spectra in the dataset, in order to reject out-of-band noise. Then, we use soft thresholding [17] to remove in-band noise in each Doppler spectrum (step 4 in Fig. 1). The soft thresholding is performed by keeping the k largest values and pad the remaining ones to 0. We choose k heuristically by considering that more than half of the Doppler

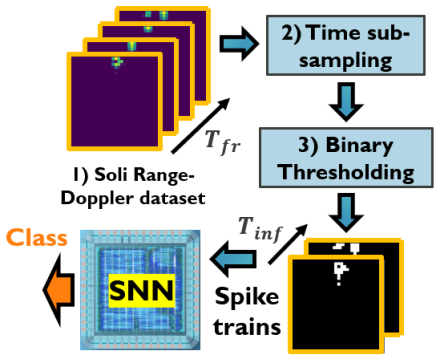

Fig. 2. Range-Doppler radar-SNN architecture proposed to solve the 12-class dataset of [4] with 91% of accuracy.

samples within the normalized frequencies $[-0.26, 0.26]$ are significant (i.e., not noise), which leads to the choice of $k = \lfloor ((192 \times (0.26 - (-0.26)))/2) \rfloor - 1 = 48$ (we tried other k values around 48, but did not observe any significant boost in SNN accuracy). After step 4, Fig. 1 shows an example radar map, resulting from the μDoppler preprocessing described above.

The preprocessed radar μDoppler maps must then be converted into event streams to be compatible with the spiking nature of our SNN. Each pixel of the map is coded as a spike train of length T_{inf} (number of time steps per inference). We encode each pixel using time-to-first-spike (TTFS) encoding [18] (step 5 in Fig. 1), where a normalized pixel of value $v \in [0, 1]$ is quantized into an event train containing only one spike located at index $T_{\text{inf}} - \lfloor v T_{\text{inf}} \rfloor$. Thus, the higher the normalized pixel value, the smaller the time to first spike. If the pixel is equal to 0, then no spikes are emitted. As we aim at low-latency inference, we choose $T_{\text{inf}} = 4$ time steps.

B. 12-Class Soli Dataset and Preprocessing

The soli dataset [4] has been acquired using a 60-GHz FMCW radar and is composed of 12 classes with a total of 5500 CFAR-processed range-Doppler magnitude acquisitions. Each gesture acquisition is a collection of maps $\text{RD}[t, l, m]$, where t is the frame index, l is the range index, and m is the Doppler index (see step 1 in Fig. 2), with a varying number of time steps $t \in [1, T_{\text{fr}}]$ per acquisition. First, we average and subsample each gesture acquisition $\text{RD}[t, l, m]$ (with varying T_{fr}) along t (step 2 in Fig. 2) to a fixed number $T_{\text{inf}} < T_{\text{fr}}$ $\forall T_{\text{fr}}$ of frames per acquisition as follows:

$$\text{RD}[n, l, m] = \frac{T_{\text{inf}}}{T_{\text{fr}}} \sum_{t=n}^{n+\frac{T_{\text{fr}}}{T_{\text{inf}}}} \text{RD}[t, l, m] \tag{3}$$

where n is the subsampled time index. Then, the resulting frames are converted to binary images $\text{RD}_b[n, l, m]$ by thresholding against 0 (step 3 in Fig. 2). Therefore, for any pixel coordinate (l^*, m^*), $\text{RD}_b[n, l^*, m^*]$ represents a spike train of length T_{inf}, set to 28 (minimum T_{fr} in the dataset). It must be noted that the use of each preprocessing methods is highly dependent on the radar used. In the 8-GHz case, the range resolution is low and the gestures remain in the same range bin. In this case, we experimentally observed that μDoppler preprocessing was systematically outperforming range-Doppler.

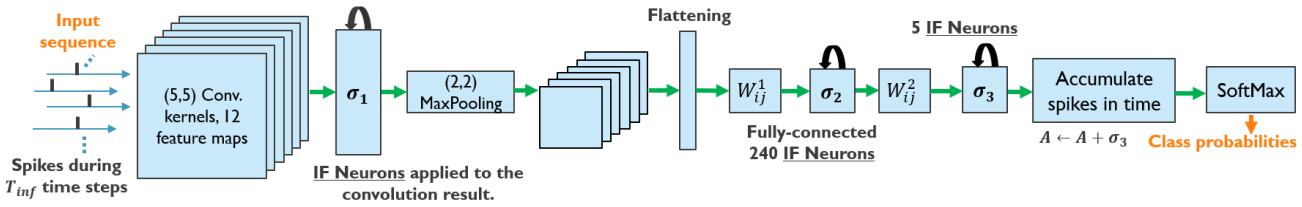

Fig. 3. SNN architecture used for radar processing. Each spiking map slice corresponding to each time step is fed one by one to the network and the IF neurons change state according to their self-recurrence (as denoted by the recurrence arrows).

C. Spiking Neural Network for Classification

To classify the spiking radar tensors, we use the architecture shown in Fig. 3 with integrate and fire (IF) neurons

$$\begin{cases} V^{k+1} = V^k + J_{\text{in}} \text{ and } S = 0, & \text{if } V^k < 1 \\ V^{k+1} = 0 \text{ and } S = 1, & \text{if } V^k \geq 1 \end{cases} \tag{4}$$

where $V^k \geq 0$ is the neural membrane potential at time step k, J_{in} is the neuron input, and S is the spiking output. As the derivative of spikes as a function of the membrane potential is ill-defined, we create a custom neuron model using the pyTorch framework [19], which behaves as (4) in forward pass. For the backward pass, we approximate the derivative using a Gaussian function (5) as the surrogate derivative [20]. In contrast to previous works, this enables the use of quantization-aware backpropagation in the spiking domain

$$\sigma'(V) \approx \frac{1}{\sqrt{2\pi}} e^{-2V^2}. \tag{5}$$

The layer-by-layer description of our SNN architecture (Fig. 3) is the following. After the spike train encoding of the radar maps, we use a $(5, 5, 12)$ convolutional layer. At each time step, the convolution result is fed to the IF neuron layer σ_1. Then, the spiking tensor at the output of σ_1 is downsampled via MaxPooling and the resulting tensor is flattened to a 1-D spiking vector. Then, two fully connected spiking layers are used and the 12-D or 5-D output of σ_3 (corresponding to the 12 or 5 gesture classes) is accumulated over time in a vector A. Finally, A is transformed via SoftMax into class probabilities. Our network architecture search was conducted with the objective of achieving a $>90\%$ accuracy with heavily quantized weights (at most 6-bit) and a small network size. For training, we use the Adam optimizer [21] with a learning rate of 10^{-3}. The batch size is 128 and the SNN is first trained for 14 epochs with full-bit weights and 1 epoch with quantized weights in the forward pass and full-bit weights in the backward pass (found through validation procedure). The accuracy of our SNN is assessed using the sixfold cross validation.

III. Experimental Results

Table I reports the performance of our proposed system (entry 6 for the 8-GHz dataset and entry 7 for the Soli dataset) against the state of the art. We evaluate the energy per classification E_c of our SNN using the hardware metrics of the μBrain SNN chip, as described in [7]

$$E_c = N_{\text{spikes}} \times E_{\text{dyn}} + \delta T \times P_{\text{stat}} \tag{6}$$

TABLE I

Our Proposed System Compared to the State of the Art. N_c is the Number of Classes, E_c is the Energy Consumption per Classification (Not Reported for Entries 2–4) and N_{BITS} Is the Number of Bits for the Network Weights (F: Float and I: Integer)

Architecture	N_c	Accuracy	E_c	N_{bits}
1) DNN [6]	12	94%	330 mJ	32-f
2) SNN-STDP [10]	8	85%	-	32-f
3) SNN-conv [9]	4	98.5%	-	32-f
4) SNN-RF [8]	11	98%	-	4-i & 32-f
5) SNN-conv [7]	4	93.4%	340 nJ	4-i
6) This work (8-GHz)	5	$93 \pm 2\%$	351 nJ	4-i
7) This work (Soli)	**12**	$91 \pm 1\%$	2 μJ	6-i

where N_{spikes} is the maximum number of spikes during classification, $E_{\text{dyn}} = 2.1$ pJ is the energy per spike, $P_{\text{stat}} = 73$ μW is the static leakage power, and δT is the inference time. Even though a smaller δT can be reached by adjusting the bias voltages that control the delay cells in [7], we assume $\delta T = 4$ ms for the 8-GHz dataset ($T_{\text{inf}} = 4$) and $\delta T = 28$ ms for Soli ($T_{\text{inf}} = 28$) to provide an upper bound on E_c.

Out of the implementation-ready SNNs using quantized weights only (entries 5–7 in Table I), our work outperforms entry 5 by $+1$ N_c versus 6) up to $\times 3$ higher number of gesture classes while not compromising much on accuracy, N_{bits} and E_c. All other entries in Table I either rely on DNNs (entry 1) and RF (entry 4), being ill-suited for ultralow-power IoT, or do not quantize their weights (entries 2 and 3, giving unclear performance in real-world SNN hardware). In addition, our work achieves a recognition accuracy within 1%–3% of the DNN in entry 1 [6] while consuming more than two orders of magnitude less energy per inference. Entries 6 and 7 show how our system trades off N_c, E_c, and N_{bits} for a target accuracy of $>90\%$. Finally, recent work sheds light on the efficacy of low-power empirical feature-based approaches [22], motivating the need for comparative investigations in future work.

IV. Conclusion

This letter has presented a novel radar-SNN architecture for ultralow-power radar gesture recognition, outperforming existing implementation-ready SNNs in terms of classification, energy, and bit-width tradeoff. The presented approach has reported several key innovations compared to previous radar-SNN systems, such as a novel radar-SNN training strategy and radar to spike encoding approaches. In order to explore further generalization of our methods, radar-SNN evaluation code has also been provided, which helps lighting the way for the emerging area of SNN-based radar processing.

REFERENCES

[1] A. Chin *et al.*, "Stability of SARS-CoV-2 in different environmental conditions," *Lancet Microbe*, vol. 1, no. 1, p. e10, 2020, doi: 10.1016/S2666-5247(20)30003-3.

[2] J. Rimmelspacher, R. Ciocoveanu, G. Steffan, M. Bassi, and V. Issakov, "Low power low phase noise 60 GHz multichannel transceiver in 28 nm CMOS for radar applications," in *Proc. IEEE Radio Freq. Integr. Circuits Symp. (RFIC)*, Aug. 2020, pp. 19–22, doi: 10.1109/RFIC49505.2020.9218297.

[3] Y.-H. Liu *et al.*, "A 680 μW burst-chirp UWB radar transceiver for vital signs and occupancy sensing up to 15 m distance," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2019, pp. 166–168, doi: 10.1109/ISSCC.2019.8662536.

[4] S. Wang, J. Song, J. Lien, I. Poupyrev, and O. Hilliges, "Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum," in *Proc. 29th Annu. Symp. User Interface Softw. Technol.*, 2016, pp. 851–860.

[5] E. Hayashi *et al.*, "RadarNet: Efficient gesture recognition technique utilizing a miniature radar sensor," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, May 2021, pp. 1–14.

[6] Y. Sun, T. Fei, X. Li, A. Warnecke, E. Warsitz, and N. Pohl, "Real-time radar-based gesture detection and recognition built in an edge-computing platform," *IEEE Sensors J.*, vol. 20, no. 18, pp. 10706–10716, May 2020, doi: 10.1109/JSEN.2020.2994292.

[7] J. Stuijt, M. Sifalakis, A. Yousefzadeh, and F. Corradi, "μBrain: An event-driven and fully synthesizable architecture for spiking neural networks," *Frontiers Neurosci.*, vol. 15, p. 538, May 2021.

[8] I. J. Tsang, F. Corradi, M. Sifalakis, W. Van Leekwijck, and S. Latré, "Radar-based hand gesture recognition using spiking neural networks," *Electronics*, vol. 10, no. 12, p. 1405, Jun. 2021.

[9] M. Arsalan, M. Chmurski, A. Santra, M. El-Masry, R. Weigel, and V. Issakov, "Resource efficient gesture sensing based on FMCW radar using spiking neural networks," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Jun. 2021, pp. 78–81.

[10] D. Banerjee *et al.*, "Application of spiking neural networks for action recognition from radar data," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Glasgow, U.K., Jul. 2020, pp. 1–10, doi: 10.1109/IJCNN48605.2020.9206853.

[11] C. Frenkel, M. Lefebvre, J. Legat, and 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," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 1, pp. 145–158, Feb. 2019, doi: 10.1109/TBCAS.2018.2880425.

[12] S. Moradi, N. Qiao, F. Stefanini, and G. Indiveri, "A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (DYNAPs)," *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 1, pp. 106–122, Feb. 2018, doi: 10.1109/TBCAS.2017.2759700.

[13] M. Davies *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, Jan. 2018, doi: 10.1109/MM.2018.112130359.

[14] C. V. Chen, *Radar Micro-Doppler Signatures: Processing and Applications*. Edison, NJ, USA: IET, 2014.

[15] F. J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," *Proc. IEEE*, vol. 66, no. 1, pp. 51–83, Jan. 1978, doi: 10.1109/PROC.1978.10837.

[16] J. B. Allen, "Short term spectral analysis, synthesis, and modification by discrete Fourier transform," *IEEE Trans. Acoust., Speech Signal Process.*, vol. ASSP-25, no. 3, pp. 235–238, Jun. 1977, doi: 10.1109/TASSP.1977.1162950.

[17] H. Xu, Z. Wang, H. Yang, D. Liu, and J. Liu, "Learning simple thresholded features with sparse support recovery," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 4, pp. 970–982, Apr. 2020, doi: 10.1109/TCSVT.2019.2901713.

[18] B. Rueckauer and S.-C. Liu, "Conversion of analog to spiking neural networks using sparse temporal coding," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Florence, Italy, May 2018, pp. 1–5, doi: 10.1109/ISCAS.2018.8351295.

[19] A. Paszke *et al.*, "Automatic differentiation in PyTorch," in *Proc. NIPS*, 2017, pp. 1–4.

[20] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks," *IEEE Signal Process. Mag.*, vol. 36, no. 6, pp. 51–63, Nov. 2019, doi: 10.1109/MSP.2019.2931595.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–15.

[22] A. Ninos, J. Hasch, and T. Zwick, "Real-time macro gesture recognition using efficient empirical feature extraction with millimeter-wave technology," *IEEE Sensors J.*, vol. 21, no. 13, pp. 15161–15170, Jul. 2021, doi: 10.1109/JSEN.2021.3072680.