End-to-End Spiking Neural Network for Speech Recognition Using Resonating Input Neurons

Daniel Auge[1](✉), Julian Hille[1,2], Felix Kreutz[3], Etienne Mueller[1], and Alois Knoll[1]

[1] Department of Informatics, Technical University of Munich, Munich, Germany
daniel.auge@tum.de, knoll@in.tum.de
[2] Infineon Technologies AG, Munich, Germany
[3] Infineon Technologies Dresden GmbH & Co., KG, Dresden, Germany

Abstract. The growing demand for complex computations in edge devices requires the development of algorithms and hardware accelerators that are powerful while remaining energy-efficient. A possible solution are spiking neural networks, as they have been demonstrated to be energy-efficient in several data processing and classification tasks when executed on specialized neuromorphic hardware. In the field of speech processing, they are especially suited for the online classification of audio streams due to their strong temporal affinity. However, so far, there has been a lack of emphasis on small-scale networks that will ultimately fit into restricted neuromorphic implementations. We propose the use of resonating neurons as an input layer to spiking neural networks for online audio classification to enable an end-to-end solution. We compare different architectures to the established method of using mel-frequency-based spectral features. With our approach, spiking neural networks can be directly used without additional preprocessing, thereby making them suitable for simple continuous low-power analysis of audio streams. We compare the classification accuracy of different network architectures with ours in a keyword spotting benchmark to demonstrate the performance of our approach.

Keywords: Spiking neural networks · Speech processing · Keyword detection

1 Introduction

Keyword spotting, as part of speech recognition, is widely used in embedded systems for a wide range of voice-activated assistants. A detector for this purpose can be operated in an always-on mode; therefore, in addition to the recognition rate, energy efficiency is a decisive factor for evaluating a detection system. Another consideration is the detector's ability to perform the desired action in real-time.

I. Farkaš et al. (Eds.): ICANN 2021, LNCS 12895, pp. 245–256, 2021.
https://doi.org/10.1007/978-3-030-86383-8_20

Current implementations consist of multiple cascaded detectors of increasing complexity to cope with these requirements. Such detectors range from simple threshold switches over classical algorithmic signal processing to complex neural networks. The growing demand for smart devices and their capabilities expects even better performance with further improved energy efficiency. Many embedded artificial neural network (ANN) architectures have been proposed to resolve this [3,26]. Ideally, also large-scale speech recognition should be performed directly in the edge device. Due to high complexity, this task is currently offloaded to cloud servers.

Recently, researchers have demonstrated promising results in efficient signal processing and speech recognition using spiking neural networks (SNNs) [18,23]. SNNs, which can be operated on dedicated neuromorphic hardware, show a significantly lower energy consumption than comparable classical ANNs [5,8,17]. They do so by exchanging short pulses in the time domain, "spikes", instead of static continuous-valued activations. Accordingly, energy is consumed only during the update of a neuron whenever a spike arrives at its input.

In this work, we compare different modeling approaches for simulating SNN behavior. Simultaneously, we use different neuron behaviors and connectivity strategies to identify the most suitable network for an end-to-end keyword spotting on restricted hardware. Therefore, we demonstrate resonating neurons as input layer to transform an audio signal into a frequency selective spatiotemporal spike representation. Thus, the network can perform keyword detection solely with spiking neurons without using digital signal processing such as mel-frequency cepstral coefficients (MFCC). In a neuromorphic realization, an analog electrical signal of a microphone can be directly fed into resonating neurons. This solution not only saves energy by turning off the digital logic, including the analog-to-digital converter, until a keyword is recognized by the neural circuit, but it also adds an extra layer of privacy for an end-user.

2 Related Work

Most modern speech recognition systems are based on non-spiking ANNs. They use recurrent or convolutional network architectures to detect spoken words in audio signals [1,10]. For this purpose, the input signal is divided into windowed blocks and a short-time Fourier transform is applied, resulting in a spectrum that changes over time. Typically, the spectrum is then projected to the mel-frequency scale and serves, along with its first and second temporal derivative, as the input feature vector.

Other neural network-based approaches exist, which directly analyze an audio stream without prior feature generation. The network learns feature extraction from the ground up while still operating on fixed-sized windows of input data. The proposed deep and convolutional architectures, therefore, exceed millions of trainable parameters and result in large networks [12–14,20].

Early works on biologically plausible audio processing solutions based on SNNs demonstrated small, energy-efficient networks, that show stimulus-specific network activities when stimulated with simple stimuli [21]. Then, an artificial cochlea [7] was used to transform the audio signal into a spiking representation.

With recent advances in supervised gradient descent-based learning algorithms for SNNs [4,16], spiking networks that perform keyword spotting in the spiking domain have been proposed [18]. Especially, Blouw and Eliasmith [5,6] focus on potential energy savings during this task using these biologically inspired networks on specialized neuromorphic hardware. They report up to 10x energy savings using low-energy neural network accelerators on the Loihi chip [8] or the upcoming SpiNNaker2 chip [15] compared to current ANN-based approaches. For the representation of input data as a sequence of events/spikes, different encoding methods can be applied. The calculated MFCC can be interpreted as the amplitude of a current that can be used to excite an input neuron [23,24]. Another similar approach is to interpret the amplitudes as a spike rate, which is commonly used for converting an ANN to an SNN [5].

3 Methods

Spiking neurons exchange information in the form of short all-or-nothing action potentials. Each neuron emits a spike as soon as the value of its hidden variable V reaches a certain firing threshold θ. In analogy to their biological counterpart, this hidden value is called membrane voltage. The membrane is charged whenever an action potential reaches the neuron via a connection between two neurons called a synapse. Similar to classical ANNs, these connections have a specific weight, which can be adapted during training to fit the desired behavior.

3.1 Spike Encoding

For a spiking network to process data, the input has to be translated into spike events. In a recent study, it has been shown that resonating neurons can be used to perform a spectral analysis on analog input data [2]. Multiple frequency-tuned resonate-and-fire [11] neurons can be used as a filter bank that emits spikes with a rate proportional to the power density of the analyzed frequencies. With that, spectral analysis and conversion into spikes are performed simultaneously. In addition, we achieve a high temporal resolution since no sliding window is needed, as opposed to a fixed-length Fourier transform.

The differential equations describing the resonating neuron are given by

$$\dot{y} = -y\,d - 2\pi f_0\,v + i(t) \tag{1}$$

$$\dot{v} = -v\,d + 2\pi f_0\,y. \tag{2}$$

This specialized neuron comprises of two coupled membranes y and v to enable the resonating behavior. Here, y and v describe the voltage and current-like variables of the neuron with its resonant frequency f_0. d is a damping value which leads to an exponential decay of the state variables over time. The input of the system is given by the current $i(t)$, which can be any arbitrary time-dependent signal or spike train, as initially proposed by Izhikevich [11]. An output spike z is generated as soon as the voltage-like variable v surpasses its firing threshold v_{th}:

$$z = \begin{cases} 1, & \text{if } v > v_{\mathrm{th}} \rightarrow y = 0, v = 0, v_{\mathrm{th}} = v_{\mathrm{th}} + v_{\mathrm{th}} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

After each output spike, the state variables are reset and the threshold is increased to achieve a spike rate adaption depending on the amplitude of the signal's spectral components. The threshold can be adapted linearly or exponentially, but an exponential adaption, as shown in Eq. 3, showed the best results in this application. The firing threshold itself also experiences an exponential damping

$$\dot{v_{\mathrm{th}}} = (v_{\mathrm{th},0} - v_{\mathrm{th}})\, d, \tag{4}$$

which ensures a weak upper boundary and a reset over time. With that, the neuron can adapt to a large range of signal amplitudes (see Fig. 1).

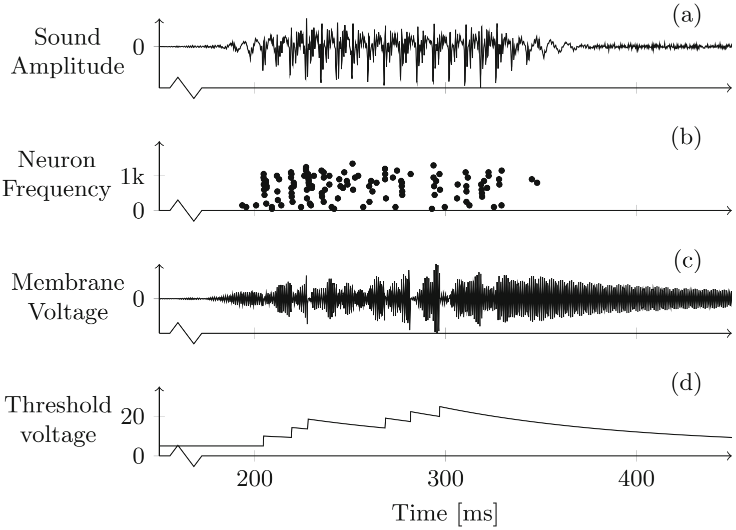


Fig. 1. Simulation of the resonating input layer using an exemplary audio sample. (a) The waveform of the audio sample. (b) Spike events generated by 40 resonating neurons. Their resonance frequencies are linearly spaced between 0 and 2,000 Hz. (c, d) Membrane voltage and threshold adaption of one resonating neuron.

As a result, a spike emitted by a resonating neuron contains a combination of different information: The spike signalizes the presence of the associated resonance frequency within the signal. In addition, the number and exact timing of the spikes encodes the amplitude of the spectral component and its development over time. With the precise interaction of input stream, membrane resonance, spike emission, and threshold adaption, a unique spatiotemporal spike train is created which is analyzed by the succeeding populations of spiking neurons.

3.2 Neuron Models

So far, a variety of neuron models have been proposed for the use in SNNs in the literature. They range from highly realistic models, which can replicate complex biological behavior of single neurons, to the simplest models, which offer only an abstraction of the biological inspiration but are suited for simulating large networks of neurons.

The base model used in this work is the leaky-integrate-and-fire (LIF) neuron with the membrane potential

$$V[t_n] = \alpha V[t_{n-1}] + (1-\alpha)\, I[t_n] - I_{\text{reset}}. \tag{5}$$

The parameter $\alpha = e^{-1/\tau}$ describes the exponential decay of the membrane voltage over time, whereas τ describes the time constant of the neuron. The input charge current I consists of the sum of weighted spike inputs S_j:

$$I_i[t_n] = \sum_j W_{ij} S_j[t_{n-1}]. \tag{6}$$

The reset current

$$I_{\text{reset}} = Z[t_{n-1}]\,\theta \tag{7}$$

is subtracted if the neuron emitted an output spike

$$Z[t_n] = \begin{cases} 1, & \text{if } V[t_n] > \theta \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

in the previous time step. By achieving the reset-by-subtraction of the threshold voltage, the information of an extremely high activation is preserved for the next time steps. Therefore, it is highly probable that a further spike will be emitted soon. A reset of the membrane potential to zero, on the other hand, would discard this information, potentially resulting in a higher ability to generalize. In our experiments, however, we consistently achieved higher performances using the reset-by-subtraction scheme.

To enable further communication within a neuron population, recurrent connections can be established. The index k indicates a presynaptic neuron within the same population as neuron i. By introducing the recurrent weight matrix R, the charging current of each neuron can, therefore, be extended to

$$I_i[t_n] = \sum_j W_{ij} S_j[t_{n-1}] + \sum_k R_{ik} S_k[t_{n-1}]. \tag{9}$$

With that, the membrane potential of a neuron depends on the activations within the same neuron population, which can also be viewed as lateral connections. Thus, the potential memorization capability of the population is increased.

3.3 Learning

Surrogate Gradient Descent: The spike emission operation shown in Eq. 8 is not differentiable and is, therefore, not suited for gradient descent-based learning methods. However, the use of surrogate gradients to enable learning is being consolidated in the development of SNNs [4,16]. The surrogate gradient ψ used in this work is

$$\psi = \max\left(1 - \left|\frac{v}{\theta} - 1\right|, 0\right).$$

On this basis, the gradient is determined by the relationship between the membrane potential and the threshold voltage, rather than the spike event.

To apply this gradient and backpropagate the error through the network and time, the network has to be simulated in discrete time steps. Therefore, we separate the input layer from the rest of the network. For the input layer, it is crucial to exactly calculate the differential equations. The following layers in contrast need to be discretized in time to apply the learning algorithm. Since there is no recurrent connection to the input of the network, the layers can be separated and simulated independently. This, however, is only a limitation during learning.

Time Constant Learning: In addition to the synaptic weights, the time constant τ (see Eq. 5) can be adapted to tune the temporal behavior of the network [25]. It controls the neuron's membrane voltage leakage over time. A large time constant leads to a small voltage leakage, enabling the long retention of information about past input spikes. A small time constant, on the other hand, enables short-term coincidence detectors, without being biased by the recent input spike history.

4 Experiments

Different network architectures are evaluated using the Speech Commands dataset [22] consisting of 65,000 audio recordings of known commands, unknown words, and silence. In a preprocessing step described by the author of the dataset, the recordings are superimposed with background noise at random volume levels.

The raw audio stream is encoded into a spike representation using resonating neurons. Following the standard of using 40 MFCC, we encode the input with 40 resonating neurons. This is simulated using exact solving of the coupled differential equations describing the resonating neurons. The resulting spikes serve as the input to the next stage of the network, which is simulated in discrete time steps to enable backpropagation learning. This part of the network is subject to optimization and architecture search. The output layer consists of non-spiking integrators – one for each class – which are evaluated after each training example to calculate the respective loss and accuracy values. The resulting abstract network architecture is shown in Fig. 2. In additions, we implement a standard MFCC-based preprocessing similar to [18] and compare it with our approach. In this case, mel-frequency features are directly applied to the first hidden layer of the network.

During the experiments, we examine the performance of three main architectures: simple feedforward networks, networks with recurrent neuron populations, and convolutional networks. In the former two cases, we also distinguish between models with one or two hidden layers. The architecture based on convolutional networks uses only one-dimensional convolutions along the frequency axis of the input. By omitting a convolution along the temporal dimension, we emphasize the inherent temporal properties of spiking neurons.

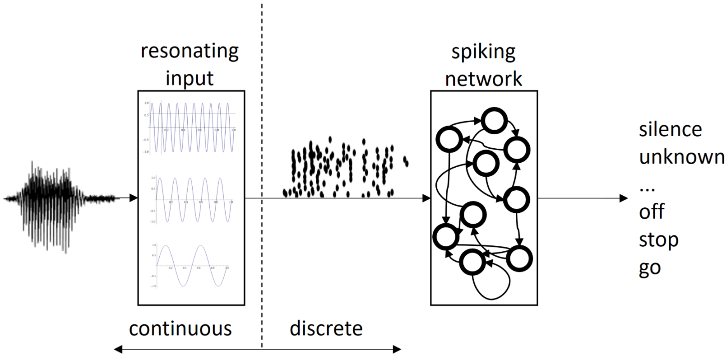


Fig. 2. Architecture of the evaluated system. The waveform is directly fed into the network without preprocessing. The resulting spikes produced by the input layer are propagated through the network. The output neuron with the highest membrane voltage selects the inferred class.

4.1 Results

In the first setup, we evaluate the relationship between the classification accuracy and the number of tunable variables. An exemplary extract of the neuron activity during the evaluation of both networks is depicted in Fig. 3. The sparseness of spike activations in the input and hidden layer can be seen. For better comparability, we chose networks consisting of only one population to avoid ambiguous distributions of the limited number of connections between multiple populations. Figure 4a shows the achieved accuracies for a simple feedforward population and a population with recurrent connections, both using RF neurons as input encoders. Note that due to the different connection schemes, the two populations do not share the same total number of neurons involved at equal numbers of variables. When the number of variables exceeds 20,000, the achieved accuracy begins to saturate. Nontheless, the difference of 10% points between the two architectures remains constant. Figure 4b depicts a confusion matrix showing the classification results of a network with recurrent neurons and a total of

40,000 tunable parameters. Class labels 0 to 11 correspond in ascending order to {silence, unknown, yes, no, up, down, left, right, on, off, stop, go}. The matrix shows a high misclassification rate for the unknown class and between word pairs {up, off} and {no, go}.

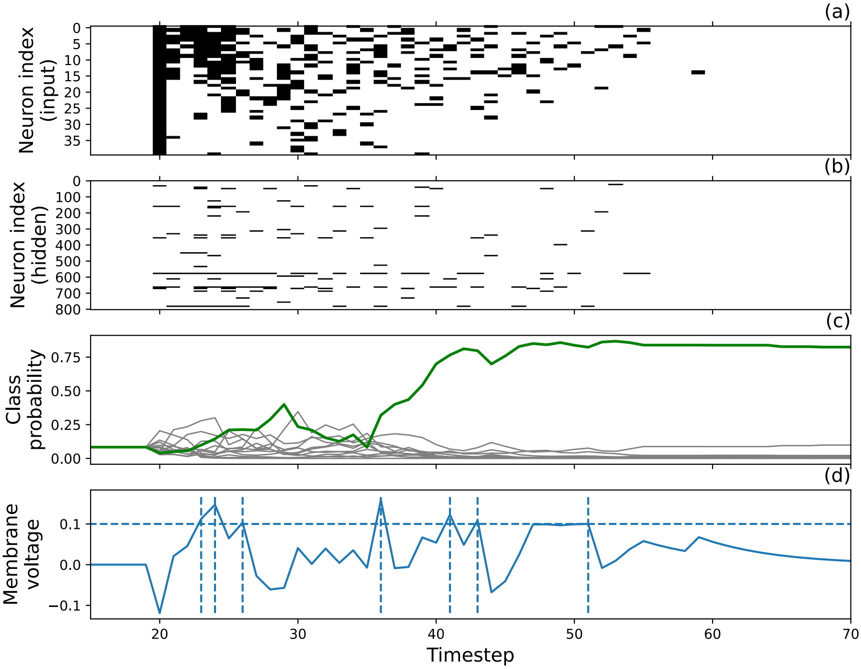
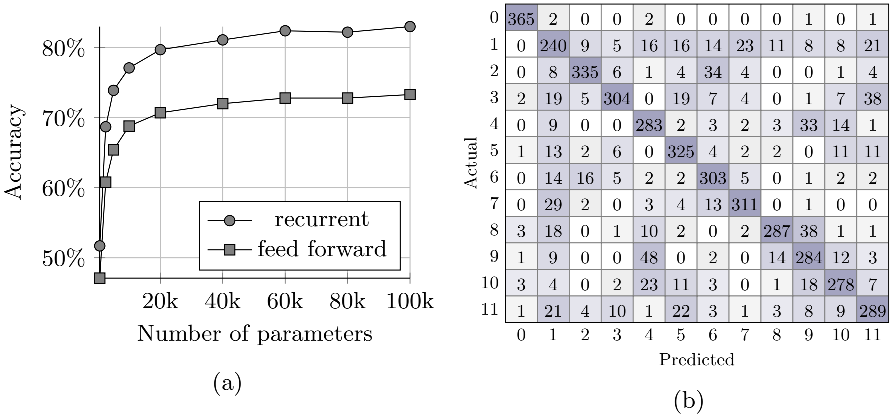


Fig. 3. Exemplary evaluation of a command. (a) Spike encoding of the audio stream using resonating neurons. (b) Spikes emitted by neurons in the hidden layer. (c) Membrane voltage of the output layer indicating the classification. (d) Membrane potential of one exemplary hidden neuron. The horizontal dashed line represents the firing threshold, the vertical lines the time instances of the spike emission.

Based on the results of the preceding experiment, we chose 40,000 trainable parameters as the common parameters of the following simulations. With that the results remain comparable while providing the intended insights about the relations between the classification accuracy, the chosen input, and the complexity of possible neuromorphic realizations. Table 1 shows the results of the evaluated architectures.

The densely connected feedforward architectures demonstrate a basic ability to solve the keyword recognition task. The better performance of multi-layered networks also underlines the common conception of the importance of deep structures [9]. In our experiments, however, increasing the number of layers further did not improve performance.



	0	1	2	3	4	5	6	7	8	9	10	11
0	365	2	0	0	2	0	0	0	0	1	0	1
1	0	240	9	5	16	16	14	23	11	8	8	21
2	0	8	335	6	1	4	34	4	0	0	1	4
3	2	19	5	304	0	19	7	4	0	1	7	38
4	0	9	0	0	283	2	3	2	3	33	14	1
5	1	13	2	6	0	325	4	2	2	0	11	11
6	0	14	16	5	2	2	303	5	0	1	2	2
7	0	29	2	0	3	4	13	311	0	1	0	0
8	3	18	0	1	10	2	0	2	287	38	1	1
9	1	9	0	0	48	0	2	0	14	284	12	3
10	3	4	0	2	23	11	3	0	1	18	278	7
11	1	21	4	10	1	22	3	1	3	8	9	289

Fig. 4. (a) Classification accuracy of networks with different sizes and connectivity rules. The x-axis describes the number of tuneable parameters in the network. The networks consist of a single hidden population. (b) Confusion matrix for a network with 40,000 parameters with recurrent connections.

Table 1. Network architectures tested with inputs consisting of mel cepstral coefficients or spikes generated by resonate-and-fire neurons. Each network consists of 40k trainable parameters.

Connection type	Architecture	MFCC	RF
Feedforward	780 neurons	72.1%	70.4%
Feedforward	175 × 175 neurons	84.1%	72.9%
Recurrent	178 neurons	84.7%	80.2%
Recurrent	105 × 105 neurons	86.7%	80.7%
Convolutional	Kernel size: 5,10; kernels: 35 × 40	86.2%	80.5%
Recurrent conv.	Kernel size: 5,10; kernels: 30 × 40	84.8%	85.5%

Among the evaluated networks with recurrent populations, the MFCC-based approaches are superior to the networks with resonating input neurons. Thus, the recurrently connected neurons can extract the temporal information present in the spectral input signal.

Networks with convolutional populations achieved the highest classification accuracies in our tests. Due to the shared weights of the convolutional kernels, it is possible to define a large number of kernels within the defined restricted number of trainable parameters. The actual size of the resulting network is significantly larger since the kernels need to be unrolled to enable parallel asynchronous processing. Including recurrent connections to the convolutional network further improved the classification accuracy.

In comparison to the evaluation results reported in the literature, our approach shows an inferior classification performance with a maximum accuracy of approximately 80%. Studies based on SNNs report error rates as low as 5.5% [18], whereas ANN-based approaches undercut this number even further [19,26]. The main differences between these works and ours are the sizes of the considered networks, their architectural design choices, and the degree of preprocessing of the analyzed data.

5 Conclusion

Our work demonstrates the successful use of small-scale SNNs with surrogate gradient descent learning and a new type of spike encoding, especially suited for online speech recognition.

Resonating neurons may be a more energy-efficient alternative to the digital processing chain used in modern speech recognition systems. Using these neurons, analog-to-digital converters, digital filters, fast Fourier transform blocks, and the MFCC feature generation can be omitted. In this work, we demonstrated that these neurons generate feature-rich spike trains that can be analyzed by the following network structures. The set of hyperparameters such as the number of resonating neurons, the choice of their resonance frequencies, or the threshold adaption characteristic leaves room for improvement and the adaption to other applications. However, the energy efficiency of this method using specialized electrical circuits remains to be proven. In addition, the classification accuracy of this approach needs to be improved to be comparable with established methods.

The network architectures considered in the experiments are particularly suitable for real-time acceleration with neuromorphic hardware since no data are buffered at any point in time, and time dependencies are represented solely by the hidden variables of neuron models or recurrent connections. Particularly, embedded systems can profit from this solution, along with the low energy consumption of SNNs on specialized hardware reported in the literature. In a hybrid realization, an SNN can serve as a low-energy always-on detector, activating a more elaborate ANN for further processing when the required activation pattern is detected.

Acknowledgments. We thank Infineon Technologies AG for supporting this research. The work is partly conducted within the KI-ASIC project that is funded by the German Federal Ministry of Education and Research (Grand Number 16ES0992K).

References

1. Abdel-Hamid, O., Mohamed, A.R., Jiang, H., Penn, G.: Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4277–4280. IEEE (2012)
2. Auge, D., Mueller, E.: Resonate-and-fire neurons as frequency selective input encoders for spiking neural networks. TUM (Technical Report) (2020)
3. Banbury, C., MicroNets: neural network architectures for deploying TinyML applications on commodity microcontrollers. arXiv preprint arXiv:2010.11267 (2020)

4. Bellec, G., Salaj, D., Subramoney, A., Legenstein, R., Maass, W.: Long short-term memory and learning-to-learn in networks of spiking neurons. In: Advances in Neural Information Processing Systems, pp. 787–797 (2018)
5. Blouw, P., Choo, X., Hunsberger, E., Eliasmith, C.: Benchmarking keyword spotting efficiency on neuromorphic hardware. In: Proceedings of the 7th Annual Neuro-Inspired Computational Elements Workshop, pp. 1–8 (2019)
6. Blouw, P., Eliasmith, C.: Event-driven signal processing with neuromorphic computing systems. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8534–8538. IEEE (2020)
7. Chan, V., Liu, S.C., van Schaik, A.: AER EAR: a matched silicon cochlea pair with address event representation interface. IEEE Trans. Circuits Syst. I Regul. Pap. **54**(1), 48–59 (2007)
8. Davies, M., et al.: Loihi: a neuromorphic manycore processor with on-chip learning. IEEE Micro **38**(1), 82–99 (2018)
9. Eldan, R., Shamir, O.: The power of depth for feedforward neural networks. In: Conference on Learning Theory, pp. 907–940. PMLR (2016)
10. Graves, A., Mohamed, A.R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6645–6649. IEEE (2013)
11. Izhikevich, E.M.: Resonate-and-fire neurons. Neural Netw. **14**(6–7), 883–894 (2001)
12. Kim, T., Lee, J., Nam, J.: Comparison and analysis of sample CNN architectures for audio classification. IEEE J. Sel. Top. Signal Process. **13**(2), 285–297 (2019)
13. Kumatani, K., et al.: Direct modeling of raw audio with DNNs for wake word detection. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp. 252–257. IEEE (2017)
14. Lee, J., Park, J., Kim, K.L., Nam, J.: Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. arXiv preprint arXiv:1703.01789 (2017)
15. Mayr, C., Hoeppner, S., Furber, S.: Spinnaker 2: a 10 million core processor system for brain simulation and machine learning. arXiv preprint arXiv:1911.02385 (2019)
16. Neftci, E.O., Mostafa, H., Zenke, F.: Surrogate gradient learning in spiking neural networks. IEEE Signal Process. Mag. **36**, 61–63 (2019)
17. Ostrau, C., Homburg, J., Klarhorst, C., Thies, M., Rückert, U.: Benchmarking deep spiking neural networks on neuromorphic hardware. arXiv:2004.01656 12397, pp. 610–621 (2020)
18. Pellegrini, T., Zimmer, R., Masquelier, T.: Low-activity supervised convolutional spiking neural networks applied to speech commands recognition. arXiv preprint arXiv:2011.06846 (2020)
19. Rybakov, O., Kononenko, N., Subrahmanya, N., Visontai, M., Laurenzo, S.: Streaming keyword spotting on mobile devices. arXiv preprint arXiv:2005.06720 (2020)
20. Sainath, T.N., et al.: Multichannel signal processing with deep neural networks for automatic speech recognition. IEEE/ACM Trans. Audio Speech Lang. Process. **25**(5), 965–979 (2017)
21. Sheik, S., Coath, M., Indiveri, G., Denham, S.L., Wennekers, T., Chicca, E.: Emergent auditory feature tuning in a real-time neuromorphic VLSI system. Front. Neurosci. **6**, 17 (2012)
22. Warden, P.: Speech commands: a dataset for limited-vocabulary speech recognition. arXiv preprint arXiv:1804.03209 (2018)
23. Wu, J., Yılmaz, E., Zhang, M., Li, H., Tan, K.C.: Deep spiking neural networks for large vocabulary automatic speech recognition. Front. Neurosci. **14**, 199 (2020)

24. Yılmaz, E., Gevrek, O.B., Wu, J., Chen, Y., Meng, X., Li, H.: Deep convolutional spiking neural networks for keyword spotting. In: Proceedings of Interspeech 2020, pp. 2557–2561 (2020)
25. Yin, B., Corradi, F., Bohté, S.M.: Effective and efficient computation with multiple-timescale spiking recurrent neural networks. arXiv preprint arXiv:2005.11633 (2020)
26. Zhang, Y., Suda, N., Lai, L., Chandra, V.: Hello edge: keyword spotting on microcontrollers. arXiv preprint arXiv:1711.07128 (2017)