Exploring Automatic Gym Workouts Recognition Locally On Wearable Resource-Constrained Devices

Sizhen Bian
PBL D-ITET
ETH Zürich
Zürich, Switzerland
sizhen.bian@pbl.ee.ethz.ch

Xiaying Wang
PBL D-ITET
ETH Zürich
Zürich, Switzerland
xiaywang@iis.ee.ethz.ch

Tommaso Polonelli
PBL D-ITET
ETH Zürich
Zürich, Switzerland
tommaso.polonelli@pbl.ee.ethz.ch

Michele Magno
PBL D-ITET
ETH Zürich
Zürich, Switzerland
michele.magno@pbl.ee.ethz.ch

2022 IEEE 13th International Green and Sustainable Computing Conference (IGSC) | 978-1-6654-6550-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/IGSC55832.2022.9969370

Abstract—**Automatic gym activity recognition on energy- and resource-constrained wearable devices removes the human-interaction requirement during intense gym sessions - like soft-touch tapping and swiping. This work presents a tiny and highly accurate residual convolutional neural network that runs in milliwatt microcontrollers for automatic workouts classification. We evaluated the inference performance of the deep model with quantization on three resource-constrained devices: two microcontrollers with ARM-Cortex M4 and M7 core from ST Microelectronics, and a GAP8 system on chip, which is an open-sourced, multi-core RISC-V computing platform from GreenWaves Technologies. Experimental results show an accuracy of up to 90.4% for eleven workouts recognition with full precision inference. The paper also presents the trade-off performance of the resource-constrained system. While keeping the recognition accuracy (88.1%) with minimal loss, each inference takes only $3.2\,\mathrm{ms}$ on GAP8, benefiting from the 8 RISC-V cluster cores. We measured that it features an execution time that is 18.9x and 6.5x faster than the Cortex-M4 and Cortex-M7 cores, showing the feasibility of real-time on-board workouts recognition based on the described data set with $20\,\mathrm{Hz}$ sampling rate. The energy consumed for each inference on GAP8 is $0.41\,\mathrm{mJ}$ compared to $5.17\,\mathrm{mJ}$ on Cortex-M4 and $8.07\,\mathrm{mJ}$ on Cortex-M7 with the maximum clock. It can lead to longer battery life when the system is battery-operated. We also introduced an open data set composed of fifty sessions of eleven gym workouts collected from ten subjects that is publicly available.**

Index Terms—**Workouts Recognition, Gym Recognition, Exercise Classification, Edge Computing, TinyML, PULP**

I. Introduction

Current commercial smart wearable devices are capable of helping people to track their fitness activities. On the other hand, they usually enable only a limited number of fitness types that can be recognized automatically without the user's input [1]. For example, the Fitbit smart wristband only identifies and records a few activities, like running and swimming. Moreover, it needs at least fifteen minutes to trigger the automatic recognition feature [2]. A popular commercial solution for recording the fitness activity is the fitness app [3], which requires user interventions like swiping and tipping on the wearable screen. These limitations are mainly due to the lack of wearable computing ability w.r.t. the complexity of body actions [4]. Since current dominant wearable motion sensing is solely based on the inertial measurement unit (IMU) [5], a typical wrist-worn smart device only captures the motion pattern of the wrist. Achieving a more smart device with onboard accurate inference capability, long lifetime, and low latency is still a challenging task, and it requires a good trade-off between energy efficiency and computational performance [4].

Automatic gym activity tracking provides the user with an intense gym experience and a reliable historical workout overview, thus improving the overall well-being of a person. A wrist-worn device could hardly sense the leg-dominated workouts, like the adductor machine, unless multiple IMUs are deployed on the limbs [6]. The non-wearable sensing modalities, like pressure mat [7], radiofrequency signal [8], and vision-based ones [9], [10], could supply much higher recognition accuracy or unlimited recognized workout types but result in user inconveniences like an extra burden or privacy issues. Another reason for wearable devices' limited workout types recognition is the limitation of local computing ability [4]. Since the success of deep neural networks in computer vision and natural language processing, researchers have explored the performance of this featuring technology on sensor-based data (which is usually one-dimensional data stream or multi-channel one-dimensional data streams from fused sensors) for human activity recognition, and got impressive result in tasks like facial expression recognition [11], daily living activities recognition [12], hand gesture recognition [13], etc. However, since the inference of deep models needs millions of multiply-accumulate operations and millions of weight and activation parameters, a wearable device usually is too weak to support the inferences regarding its memory space, clock, and computing ability.

Considering the massive deployment of Internet of Things (IoT) devices around human life, researchers have focused on skills aiming to run a neural network architecture on those resource-limited devices with kilobytes of RAM by compressing an exited model with minimal accuracy loss. For example, quantization [14], changing the model parameters from floating-point numbers to low-bit width numbers; Pruning [15], removing the least important parameters that do not affect the decision-making process of the model; Knowledge distillation [16], transferring the knowledge to the student model

978-1-6654-6550-2/22/$31.00 ©2022 IEEE

form a teacher model. Those model compression approaches can be considered individually as well as jointly [17], [18] and have shown their validity in enormous use scenarios [19]–[21]. Besides that, novel hardware architectures were also developed to bring intelligence to the edge. Such hardware platforms consume milliwatt of power and are suitable for wearable battery-operated devices [22]–[24]. Among them, the novel GAP8 architecture, which features 8 RISC-V cores, has shown impressive inference performance in certain human activity recognition tasks [4], [25]. This triggered the idea of automatic gym workout recognition and recording locally on cheap and ultra resource-constrained devices, aiming to enable a better user experience for gym enthusiasts.

This paper explored a lightweight residual tiny neural network model for gym workouts recognition on energy efficient embedded devices and demonstrated its work efficiency on three resource-constrained platforms. Following contributions are presented:

1) We first introduced a gym data set including eleven workouts for classification, collected from ten volunteers in multiple days with sensors deployed at the wrist, leg, and pocket. The data set is public available[1] aiming to promote other researchers for further exploration. The sensing modalities of the data set include the traditional inertial measurement sensors and a novel human body capacitance signal [26], which has been demonstrated with impressive alternative or complementary role for body motion sensing [27].
2) We utilized the residual neural network model and deployed the compressed model on three processors that are designed for embedded applications: the ARM Cortex-M4/M7 MCU from STMicroelectronics and the RISC-V PULP-based GAP8 MCU [28] from GreenWaves Technologies. All three platforms show minimal accuracy drop for the inferences with 8-bits weights and activation parameters.
3) We demonstrated the possibility to enable automatic activity recognition directly on-board by exploring the inference performance on the selected platforms regarding the throughput, latency, and power efficiency. While setting all three platforms with their maximum clock speed, the PULP platform takes only $3.2\,\mathrm{ms}$ for each inference, which is 18.9x and 6.5x faster than the Cortex-M4 and Cortex-M7 cores, showing the feasibility of real-time on-board workouts recognition on the described data set with 20Hz sensor data sampling rate. The multiply and accumulation (MAC) operations in each cycle also show a 13.0x and 8.1x improvement on GAP8.
4) We demonstrated the advantage of the proposed algorithm on the selected GAP8 platform in terms of energy efficiency. Experimental measurements show the energy consumed for each inference on GAP8 is $0.41\,\mathrm{mJ}$ while $5.17\,\mathrm{mJ}$ on Cortex-M4 and $8.07\,\mathrm{mJ}$ on Cortex-M7.

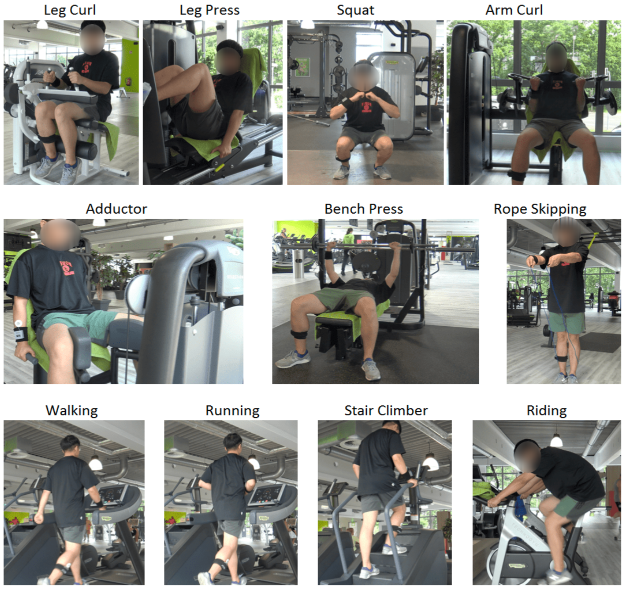

Fig. 1. Eleven gym workouts

II. Data Set and Set-up

The presented data set includes eleven popular aerobic and anaerobic workouts that are supported in a typical gym studio: Adductor, Armcurl, Benchpress, Legcurl, Legpress, Riding, Ropeskipping, Running, Squat, Stairsclimber, and Walking (as Fig. 1 depicts). Running and Walking were performed on the treadmills with the speed of 5±0.2 km/h and 8±0.5 km/h for around 2 minutes in each session. Riding and Stairclimber were done at a self-determined pace and lasted about 2 minutes. The rest exercises were trained with gym instruments (except Squat) for 3x10 repetitions. Ten volunteers participated in this study, including five females and five males. Eight of them go to the gym at least three times a week, and two of them are novices. Each participant performed the above-listed exercises for five sessions in five days.

The sensing unit contains both inertial measurement unit (IMU) and the human body capacitance (HBC) sensing unit described in [29]. Human body capacitance is a particular property describing the capacitance between the body and the ground or environment, a leg motion causes an overall body capacitance variation, and the variation could be captured by the body capacitance sensing unit worn on the wrist [30]. The two sensing units were integrated on a small PCB powered by a 3.7V lithium battery for the gym experiment data collection. The data set includes data collected with three sensor positions: the wrist, the leg, and the pocket. Since this paper only focuses on exploring local inference of the workouts, we only use the data with the sensor worn on the wrist for the purpose.

Fig. 2 depicts one whole session of IMU and HBC data from the eighth volunteer in the fourth training day with the prototype worn on the wrist. The grey color in the fourth subplot is defined as ”Null” state, which is also being classified

[1] https://github.com/zhaxidele/Toolkit-for-HBC-sensing

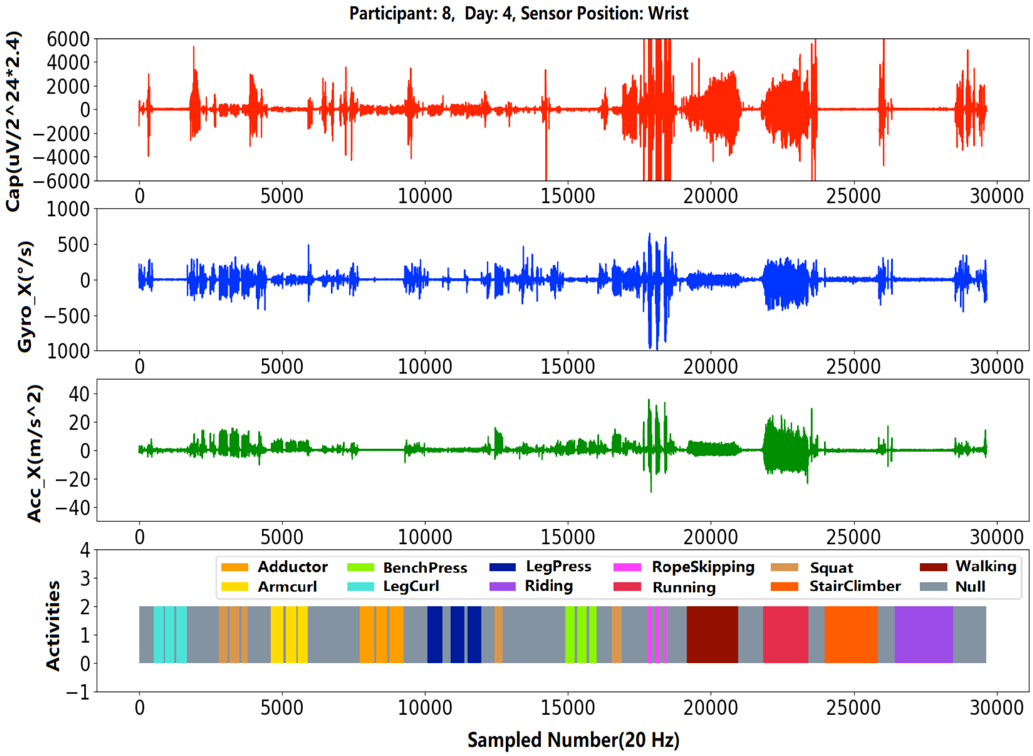

Fig. 2. Example of one session's initial measurement unit signal, body capacitance caused potential variation signal and the exercise labels, including a null class.

by the neural network model. Data was sampled with a frequency of 20 Hz. Overall, fifty sessions data from the ten volunteers were collected.

III. Proposed TinyML Residual Model

For classification, we used a residual network composed of three identity blocks, following a 1D convolutional layer and followed by a dense layer (as Fig. 3 depicts). Each identity block has three 1D convolutional layers. One dimensional CNN has been proved to be an effective way for deriving features from a fixed-length segment of the overall time series data set, where the location of the features in the segment is not so important. All convolutions in this architecture use *same* padding, which is convenient for going deeper as in our case each instance consists of a one-dimensional vector with window size (40) values. The input of the model is a seven-channel 1D data stream (7 x 40), which is normalized from the raw motion sensor data (three axes accelerometer and gyroscope, human body capacitance). The output is the predicted probability of the twelve classes (eleven workouts and one ”Null” class).

The model was trained using the categorical cross-entropy loss function and the Adam optimizer with 0.001 learning rate and 0.9 and 0.999 for β_1 and β_2, respectively. Since the data set is highly imbalanced, containing more ”Null” instances than any other exercise, every training instance is weighted based on the labels present inside the windows. The total weight of a window is inversely proportional to the frequency of its labels in the data set and is calculated based on the labels of a window.

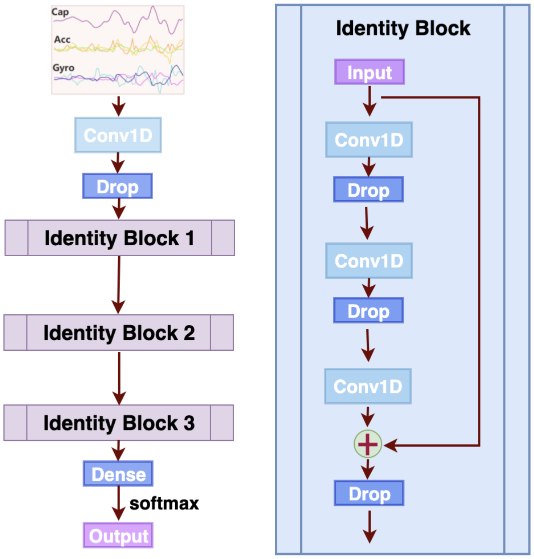

Fig. 3. Residual Deep Model

TABLE I
INFERENCE PERFORMANCE ON GAP8 AND CORTEX-M4/M7

Platform	GAP8		Cortex-M4		Cortex-M7	
MCU	1xCV32E40P+8×CV32E40P		1×Cortex M4		1×Cortex M7	
Framework	GapFlow		CUBE.AI TFLite		CUBE.AI TFLite	
Input Size	40x7		40x7		40x7	
MAC	3,051,812		3,039,408		3,039,408	
Flash	108.68kB		105.11kB		105.11kB	
RAM	33.37kB		14.41kB		14.41kB	
Accuracy(quantized)	88.1		89.3		89.3	
Accuracy(Full Precision)	90.4		90.4		90.4	
Clock	80Mhz	Max(175Mhz)	60Mhz	Max(120Mhz)	108 Mhz	Max(216Mhz)
Time/Inference[ms]	6.8	3.2	114.25	60.36	41.74	20.88
Throughput [MMAC/s]	448.80	953.70	26.60	50.35	73.05	145.57
MAC/cycle	5.610	5.450	0.44	0.42	0.68	0.67
Power [mW]	54.60	129.36	47.16	85.67	185.49	386.73
Energy/inference [mJ]	0.37	0.41	5.39	5.17	7.74	8.07
En. eff. [GMAC/s/W]	8.220	7.372	0.564	0.588	0.394	0.376

The model is trained for 1000 epochs with early stopping using patience of 100 to avoid overfitting. To show that we can learn to recognize activities across subjects, we employed a leave-one-user-out procedure where, for each fold, the test set contains all days of one subject, while the training set contains all days for the remaining ones. The TensorFlow Lite library was then used to convert the neural network model into the Tensorflow Lite file without quantization in this step. The optimization such as quantization will be performed by different frameworks specially designed for the embedded devices as described in the following subsection.

IV. HARDWARE PLATFORMS

For the evaluation of the inference performance on embedded processors, we chose three MCUs from STMicroelectronics and GreenWaves Technologies: STM32 Nucleo-144 development boards with Cortex-M4 processor and Cortex-M7 processor and GAP8, a RISC-V-based PULP platform.

The Corte-M4 platform is equipped with 2 Mbytes of flash memory, 640 Kbytes of SRAM and can operate with a frequency of up to 120 MHz. The high-performance series of Cortex-M7 is equipped with 2 Mbytes of flash memory, 512 Kbytes of SRAM, and can be operated at up to 216 MHz frequency. For the conversion from the Tensorflow Lite model to the executable files on processors, we used the CUBE.AI tool from STMicroelectronics. The tool quantizes the pre-trained machine learning models, including neural network and traditional machine learning models, to reduce memory footprint and inference time, and integrates the generated optimized library into the specific project. Users could use a graphical interface or command line interface for the automation.

The PULP platform has two compute domains: a fabric controller with one CV32E40P core and a cluster domain with 8 CV32E40P cores. The fabric controller is the central system controller and resembles a standard MCU, which can delegate compute-intensive tasks to the cluster. The fabric controller has 512 kB L2 memory extendable via a HyperBus interface. The cluster is a multi-core compute domain with a shared 80 kB directly accessible L1 memory architecture and hardware thread synchronization, and enables highly efficient, parallel implementation of algorithms giving almost optimal linear speedup. Depending on the supplied voltage level, the frequency can be boosted to 250 MHz for the fabric controller and 175 Mhz for the cluster. We used the GAPflow framework for the conversion of the Tensorflow lite file. The framework comprises two core components: the NNTool, which parses the flatbuffer file (tensorflow Lite model) into an internal network description with topology optimization, like folding the batch normalization in the convolutional weights, and performs the quantization when requested; And the AutoTiler, which computes the optimal tiling scheme and converts the parsed model into optimized, debugable C code that can be executed efficiently in parallel by the eight cores in the cluster. To increase the efficiency of the memory usage, the Autotiler also utilizes fused layers generated from the NNTool, for example, a convolution layer followed by a pooling layer and the ReLU activation function can be performed by a single layer ConvPoolRelu supported by the AutoTiler.

V. EXPERIMENTAL RESULTS

Table I presents the performance of the residual CNN model on the three hardware platforms with 8-bits quantization. Power consumption was measured by a power analyzer. The MAC operations of the proposed model are over three million and are slightly different on the platforms because of the various optimization strategy of each framework. The optimized model takes more than one hundred kilobytes of Flash space. Because of the parallel computing of the RISC-V cluster cores, the RAM space needed on GAP8 is larger than the Cortex-M platforms but still far behind the limitation (512kBytes). With the full precision inference, the workouts could be recognized with a balanced accuracy of 90.4% on unregistered users. With a quantized model, the accuracy shows a minimal drop (88.1% on GAP8 and 89.4% on ARM Cortex M4/M7). The drop on the GAP8 platform could be negligible when looking at the achieved inference performance in time and power consumption on it, as the inference speed and energy consumed on GAP8 show significant improvement over the Cortex cores. Benefiting from the 8 RISC-V CV32E40P cluster, each inference takes 6.8ms with an 80Mhz clock and 3.2ms with the maximum clock on GAP8, which is 18.9x and 6.5x faster than the Cortex-M4 and Cortex-M7 while using the full clock speed, showing the feasibility of real-time on-board workouts recognition on the described data set with 20Hz sampling rate. The fast inference could also be indicated by the MAC operations in each cycle, where GAP8 offers 5.45 MAC operations per cycle, which shows a 13.0x and 8.1x improvement over the other two platforms. The energy consumed for each inference on GAP8 is 0.41mJ while it is 5.17mJ on Cortex-M4 and 8.07mJ on Cortex-M7 with the maximum clock speed. We also changed the clock on each platform for classification by using half of the full clock speed. It shows that the inference time and power consumption increase almost linearly as the clock increases. Still, the energy efficiency stays nearly the same on each platform since the time used for each inference also linearly decreases.

VI. Conclusion

This work first presented a gym workouts data set for sensor-based activity recognition. The exploration of the automatic workouts recognition on three resource-constrained devices with a residual neural network was then described: two 32-bit microcontrollers with ARM-Cortex M4 and M7 core from STMicroelectronics and a RISC-V based open-source multi-core computing platform PULP from GreenWaves Technologies. Benefiting from the 8 RISC-V CV32E40P cluster, the inference performance on GAP8 shows significant advantages regarding the inference time, throughput, and energy efficiency over Cortex M4 and M7 cores while keeping the accuracy with minimal loss after quantization. With the maximal clock speed enabled, the GAP8 recognized the workouts with an accuracy of 88.1% with a drop of 2.3%, which is negligible considering the impressive performance improvement when compared with Cortex-M7 in inference time (3.2ms, 6.5x), throughput (953.70 MMAC/s, 6.6x), and energy efficiency(7.372 GMAC/s/W, 19.6x). This work demonstrated the feasibility of automatic gym workouts recognition directly on smart devices with limited hardware resources and battery life, and presented the outstanding inference performance on the open-source PULP platform than the classical Cortex-M platforms.

References

[1] R. Khurana, “The past, the present, and the future of fitness tracking,” *XRDS: Crossroads, The ACM Magazine for Students*, vol. 25, no. 4, pp. 30–33, 2019.

[2] W. Leung, L. Case, M.-C. Sung, and J. Jung, “A meta-analysis of fitbit devices: same company, different models, different validity evidence,” *Journal of Medical Engineering & Technology*, vol. 46, no. 2, pp. 102–115, 2022.

[3] M. Radhakrishnan, A. Misra, R. K. Balan, and Y. Lee, “Gym usage behavior & desired digital interventions: An empirical study,” in *Proceedings of the 14th EAI International Conference on Pervasive Computing Technologies for Healthcare*, 2020, pp. 97–107.

[4] X. Wang, M. Magno, L. Cavigelli, and L. Benini, “Fann-on-mcu: An open-source toolkit for energy-efficient neural network inference at the edge of the internet of things,” *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4403–4417, 2020.

[5] S. Ishii, K. Nkurikiyeyezu, M. Luimula, A. Yokokubo, and G. Lopez, “Exersense: Real-time physical exercise segmentation, classification, and counting algorithm using an imu sensor,” in *Activity and Behavior Computing*. Springer, 2021, pp. 239–255.

[6] K.-H. Chang, M. Y. Chen, and J. Canny, “Tracking free-weight exercises,” in *International Conference on Ubiquitous Computing*. Springer, 2007, pp. 19–37.

[7] M. Sundholm, “Activity recognition using floor based pressure sensors,” Ph.D. dissertation, DFKI GmbH, 2014.

[8] X. Guo, J. Liu, C. Shi, H. Liu, Y. Chen, and M. C. Chuah, “Device-free personalized fitness assistant using wifi,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 4, pp. 1–23, 2018.

[9] P. Ganesh, R. E. Idgahi, C. B. Venkatesh, A. R. Babu, and M. Kyrarini, “Personalized system for human gym activity recognition using an rgb camera,” in *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, 2020, pp. 1–7.

[10] P. Vyas, “Pose estimation and action recognition in sports and fitness,” 2019.

[11] I. M. Revina and W. S. Emmanuel, “A survey on human face expression recognition techniques,” *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 6, pp. 619–628, 2021.

[12] R. G. Hussain, M. A. Ghazanfar, M. A. Azam, U. Naeem, and S. Ur Rehman, “A performance comparison of machine learning classification approaches for robust activity of daily living recognition,” *Artificial Intelligence Review*, vol. 52, no. 1, pp. 357–379, 2019.

[13] A. Jaramillo-Yánez, M. E. Benalcázar, and E. Mena-Maldonado, “Real-time hand gesture recognition using surface electromyography and machine learning: A systematic literature review,” *Sensors*, vol. 20, no. 9, p. 2467, 2020.

[14] C. Zhu, S. Han, H. Mao, and W. J. Dally, “Trained ternary quantization,” *arXiv preprint arXiv:1612.01064*, 2016.

[15] S. Han, H. Mao, and W. J. Dally, “Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding,” *arXiv preprint arXiv:1510.00149*, 2015.

[16] C. Buciluǎ, R. Caruana, and A. Niculescu-Mizil, “Model compression,” in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 535–541.

[17] Y. Gil, J.-H. Park, J. Baek, and S. Han, “Quantization-aware pruning criterion for industrial applications,” *IEEE Transactions on Industrial Electronics*, vol. 69, no. 3, pp. 3203–3213, 2021.

[18] Y. Choi, J. Choi, M. El-Khamy, and J. Lee, “Data-free network quantization with adversarial knowledge distillation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 710–711.

[19] S. Bian and P. Lukowicz, “Capacitive sensing based on-board hand gesture recognition with tinyml,” in *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*, 2021, pp. 4–5.

[20] T. M. Ingolfsson, X. W. M. Hersche, A. Burrello, L. Cavigelli, and L. Benini, “Ecg-tcn: Wearable cardiac arrhythmia detection with a temporal convolutional network,” in *2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, 2021, pp. 1–4.

[21] X. Wang, M. Hersche, B. Tömekce, B. Kaya, M. Magno, and L. Benini, “An accurate eegnet-based motor-imagery brain–computer interface for low-power edge computing,” in *2020 IEEE international symposium on medical measurements and applications (MeMeA)*. IEEE, 2020, pp. 1–6.

[22] M. S. Louis, Z. Azad, L. Delshadtehrani, S. Gupta, P. Warden, V. J. Reddi, and A. Joshi, “Towards deep learning using tensorflow lite on risc-v,” in *Third Workshop on Computer Architecture Research with RISC-V (CARRV)*, vol. 1, 2019, p. 6.

[23] A. Garofalo, M. Rusci, F. Conti, D. Rossi, and L. Benini, “Pulp-nn: accelerating quantized neural networks on parallel ultra-low-power risc-v processors,” *Philosophical Transactions of the Royal Society A*, vol. 378, no. 2164, p. 20190155, 2020.

[24] M. Sorbaro, Q. Liu, M. Bortone, and S. Sheik, “Optimizing the energy consumption of spiking neural networks for neuromorphic applications,” *Frontiers in neuroscience*, p. 662, 2020.

[25] S. Benninger, M. Magno, A. Gomez, and L. Benini, “Edgeeye: A long-range energy-efficient vision node for long-term edge computing,” in

2019 Tenth International Green and Sustainable Computing Conference (IGSC). IEEE, 2019, pp. 1–8.

[26] S. Bian and P. Lukowicz, "A systematic study of the influence of various user specific and environmental factors on wearable human body capacitance sensing," in *EAI International Conference on Body Area Networks*. Springer, 2021, pp. 247–274.

[27] S. Bian, V. F. Rey, P. Hevesi, and P. Lukowicz, "Passive capacitive based approach for full body gym workout recognition and counting," in *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE, 2019, pp. 1–10.

[28] E. Flamand, D. Rossi, F. Conti, I. Loi, A. Pullini, F. Rotenberg, and L. Benini, "Gap-8: A risc-v soc for ai at the edge of the iot," in *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*. IEEE, 2018, pp. 1–4.

[29] S. Bian, S. Yuan, V. F. Rey, and P. Lukowicz, "Using human body capacitance sensing to monitor leg motion dominated activities with a wrist worn device," in *Sensor-and Video-Based Activity and Behavior Computing: Proceedings of 3rd International Conference on Activity and Behavior Computing (ABC 2021)*. Springer Nature, 2022, p. 81.

[30] S. Bian, V. F. Rey, J. Younas, and P. Lukowicz, "Wrist-worn capacitive sensor for activity and physical collaboration recognition," in *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2019, pp. 261–266.