tinyRadar for Gesture Recognition: A Low-power System for Edge Computing

Dileep Kankipati[1*], Madhu Munasala[1*], Dasari Sai Nikitha[1], Satyapreet Singh Yadav[1], Sandeep Rao[2], Chetan Singh Thakur[1]
{kdileep, munasalam, dasaris, satyapreets, csthakur}@iisc.ac.in, s-rao@ti.com
[1]Department of Electronic Systems Engineering, Indian Institute of Science, Bangalore, India, 560012
[2]Texas Instruments, Bangalore, India, 560093

***Abstract*—Hand gesture recognition (HGR) plays a pivotal role in improving human-machine interaction across domains like smart homes/vehicles and wearable devices. While vision-based HGR systems encounter challenges with lighting, complex backgrounds, and occlusion, radar-based systems overcome these limitations by harnessing electromagnetic principles. This paper presents tinyRadar, a real-time, low-power, single-chip radar solution for HGR. By leveraging miniaturized mmWave radar hardware, tinyRadar offers a compact and cost-effective HGR solution. The Texas Instruments IWRL6432 radar is utilized, achieving a total power consumption of less than 80mW and a memory footprint of $\sim$ 11 KB for the quantized inference model and $<$ 256 KB for the entire system. The solution utilizes quantized depthwise separable convolutions and integrates a hardware accelerator and Cortex®-M4 microcontroller for real-time inference. With its small form factor and low power requirements, tinyRadar facilitates on-edge implementation, delivering 95% real-time inference accuracy for six gestures. This paper contributes to developing wearable gadgets and IoT devices that seamlessly incorporate HGR technology.**

***Index Terms*—hand gesture recognition, IWRL6432 single-chip mmWave radar, depthwise separable convolution, edge computing, low power, velocity-time map, angle-time map**

I. Introduction

Hand gesture recognition (HGR) is a highly desirable component of human-machine interaction (HMI) systems, enabling intuitive and natural communication between humans and machines. HGR has gained extensive applications across various domains, including virtual reality (VR), augmented reality (AR), smart homes/vehicles, and Internet of Things (IoT) devices, by eliminating the reliance on traditional controls, enhancing convenience, accessibility, and the overall user experience in smart living environments [1]–[3]. The potential of HGR with advancements in machine learning (ML) has led researchers to develop novel recognition systems using different sensing modalities such as cameras, radar, ultrasound, inertial measurement unit (IMU) sensors, and many more [4]–[7].

Vision-based HGR systems leverage RGB or infra-red (IR) cameras and employ computer vision algorithms [8]. However, accurately detecting gestures poses challenges attributed to factors such as varying illumination, shadows, complex backgrounds, and spatiotemporal variations in hand postures [9]. Moreover, these systems necessitate a structured environment, entail high computational complexity and exhibit sensitivity to partial occlusion [10]. Also, vision-based machine learning is susceptible to adversarial attacks, enabling malicious visual inputs that compromise the system's integrity [11] and privacy invasion. Wearable-based HGR systems address the limitations of vision-based sensors but can result in user discomfort and restricted movement, such as when incorporating IMUs into hand gloves [12], [13].

In contrast to vision-based systems, radar-based systems leverage electromagnetic principles, enabling them to overcome challenges associated with illumination and spatiotemporal variations [14]. These systems prioritise user privacy by processing sparse point cloud data while maintaining efficient computational performance [15], [16]. Moreover, radar waves have the ability to penetrate certain materials such as curtains, paper, fog, and smoke, ensuring robust performance even in occluded environments [17]. In recent years, there has been significant progress in miniaturizing mmWave radar-based hardware [18], developing compact and cost-effective ML solutions that offer high flexibility. This makes mmWave radar optimal for implementing HGR systems.

Several research studies in HGR using mmWave radar have primarily focused on utilizing range, doppler, and angle information, either individually or in combination, for feature generation. These studies have explored various deep learning (DL) models, such as convolutional neural networks (CNN) and long short-term memory (LSTM), to improve the accuracy of gesture classification [14], [19], [20]. However, only a few of these approaches have successfully demonstrated on-edge implementation for real-time, on-device gesture recognition [19], [21], [22]. The emerging trend in HGR is integrating this technology into wearable gadgets and IoT devices. Thus, demanding compact form-factor solutions that can be deployed on microcontrollers with limited power and memory resources. Previous work by Soli [22], [23] utilized over 600 MB of memory and consumed 300 mW of power, making it unsuitable for low-memory microcontrollers and devices with limited power. In contrast, tinyRadarNN [21] improved upon this by deploying a model with $\sim$ 90 KB of memory and $<$ 120 mW power consumption. However, their approach involved separating the sensing element, mmWave radar, from

*Equal contribution

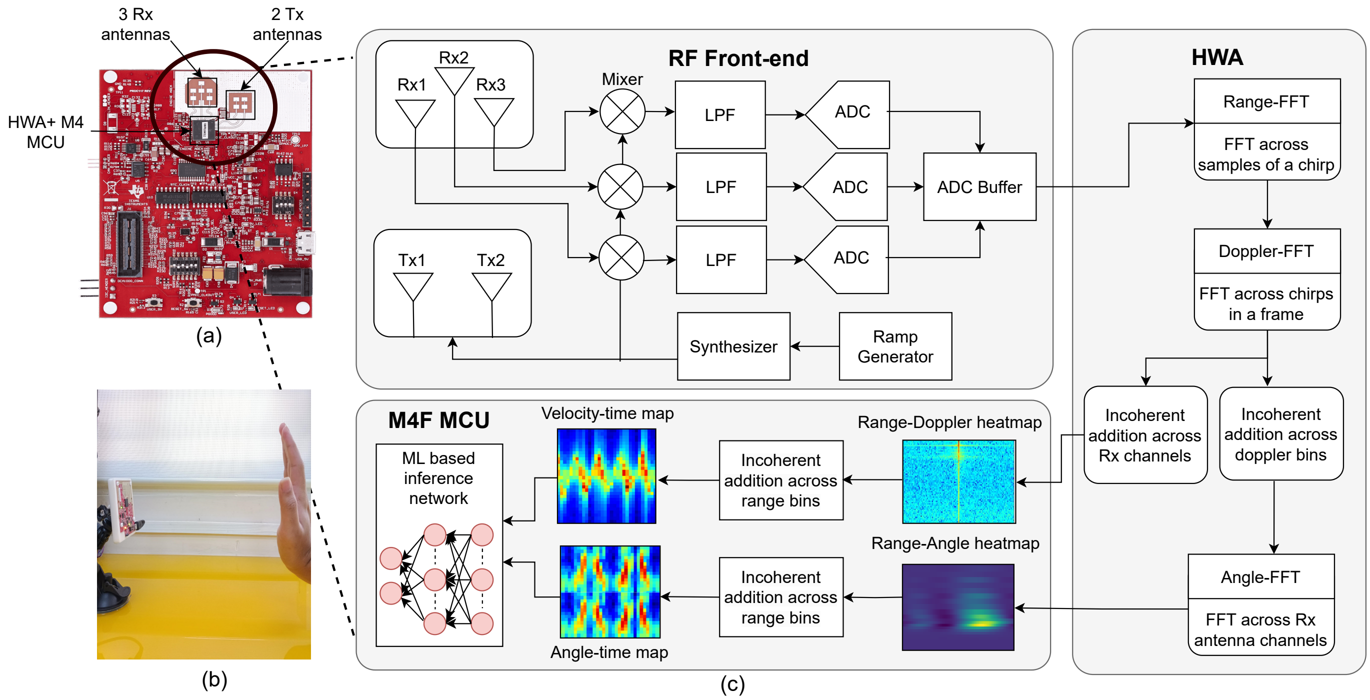

Fig. 1. Block diagram of tinyRadar based HGR system (a) IWRL6432 radar board comprising two transmit and three receive antennas along with HWA and Cortex®-M4. (b) A person performing a gesture in front of tinyRadar. (c) Block diagram indicating the signal and implementation flow from sensing the target environment using the RF front-end to VT and AT map generation and classification.

the processing element, the GAP8 development board.

We propose tinyRadar, a low-power, real-time, on-edge single-chip radar solution for HGR based on the Texas Instruments IWRL6432 radar [24]. Our solution achieves a total power consumption (including the sensor and model) of less than 80mW at 160 MHz and has a compact memory footprint of $\sim$ 11 KB for the quantized model and less than 256 KB for the entire system. Through leave-one-out cross-validation (LOOCV), we achieve real-time inference accuracy of 95% for six different gestures trained on a dataset collected from 9 users. Our solution leverages quantized depthwise separable convolutions to achieve a compact model size while harnessing the capabilities of the hardware accelerator (HWA) and Cortex®-M4 microcontroller integrated in the radar board. This integration enables the simultaneous execution of sensing and processing functions, facilitating the implementation of complex algorithmic flows in real-time while operating at low power.

This paper is organized as follows: Section II introduces the mmWave radar-based HGR system. Section III provides information about the collected dataset. Section IV outlines the signal processing flow, followed by Section V, which describes the classification model. The results are presented in Section VI, and Section VII concludes the paper.

II. System Description

Figure 1 illustrates the operation of tinyRadar, where frequency-modulated continuous wave (FMCW) signals, known as chirps, are generated using a ramp generator and transmitted through two TX antennas. The user's hand reflects the chirps during gesture performance, and three RX antennas receive the reflected signals. The radar operates in time division multiplexing multiple input multiple output (TDM-MIMO) mode, where each TX antenna sequentially transmits. After down-conversion to intermediate frequency (IF), digitization, and storage in a buffer, the received signals undergo a series of signal processing steps in the HWA, described in Section IV. These steps yield velocity-time (VT) and angle-time (AT) maps, which are fed to a quantized depthwise separable convolution network deployed on Cortex®-M4 microcontroller for real-time inference. The VT and AT maps were generated directly on the radar board for data collection and network training on six distinct gestures: slow swipe, fast swipe, push, circle, cross, and rest. The "rest" gesture indicates no activity or when the hand is stationary. Compared to our prior work [15], [16], using the TDM-MIMO technique for generating AT maps alongside VT maps boosts accuracy while minimizing memory and power consumption using a quantized depthwise separable convolution network.

III. Dataset Description

The dataset used in this study consists of recordings of hand gestures performed by 9 participants. Each participant performed six gestures. The data was collected using IWRL6432, which was programmed to generate VT and AT maps through an onboard signal processing pipeline. Figure 2 shows the VT and AT maps corresponding to each gesture. The radar board was positioned on a table of height $\sim$ 1.5m, and the participants performed the gestures at various distances, ranging up to a maximum of 1m, and varying angles from the radar while in a sitting position. Each gesture was recorded in two separate sessions, each lasting 3 minutes. A total of 324 minutes of data was collected, equivalent to 1,55,520 frames of data. The recorded data was then streamed from the radar board to a local PC for offline training. After the data cleaning process, a set of 9,680 VT and AT maps were generated, each sized 32 x 16.

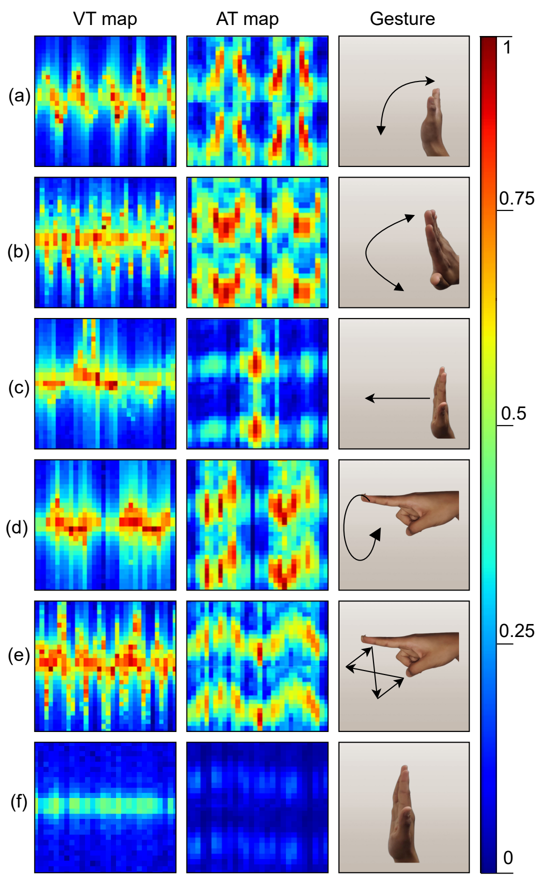

Fig. 2. Gesture types and their feature maps: Snapshots of gesture performed by the user (right) and corresponding normalized VT and AT map (left, middle), each of size 32 x 16. The gestures are as follows: (a) slow swipe (b) fast swipe (c) push (d) circle (e) cross (f) rest

The data collection process for this application involved selecting specific chirp configurations outlined in TABLE I. The choice of frequency slope and ADC sampling rate was made to enable high signal-to-noise ratio (SNR) detection up to a range of 1m. By using 128 chirps in a burst with a burst periodicity of 200μs, a velocity resolution of $\sim$ 5 cm/s was achieved, resulting in high-resolution VT maps. Sending more chirps reduces the processing time budget within a frame. To maintain adequate processing time, a 125 ms frame duration was selected. The utilization of TDM-MIMO technique allowed for the synthesis of six antenna elements, providing enhanced angular resolution ($\theta_{res} = \frac{2}{N} = 0.33$ radian) compared to the baseline configuration with only three receive antennas ($\theta_{res} = 0.67$ radian).

TABLE I
CHIRP PARAMETERS

Parameter	Value
Start frequency	60 GHz
Idle time	6 μs
Ramp end time	30 μs
Chirp frequency slope	90 MHz/μs
ADC samples	128
ADC sampling rate	5500 ksps
Bursts per frame	128
Chirps per burst	2
Burst period	200 μs
Frame duration	125 ms
Tx antennas used	2
Rx antennas used	3
Azimuth FFT size	16

IV. ONBOARD SIGNAL PROCESSING

During the data acquisition process, the HWA reads data from the ADC buffer to extract the range information of the user's hand. This is accomplished by applying the fast fourier transform (FFT) on the chirp samples, known as Range-FFT. The FFT is performed on each chirp of length 128, which is then reduced to 64 by considering only the real part. Once a complete data frame is received, the HWA applies FFT across the chirps of the frame, referred to as Doppler-FFT. Incoherent addition is performed across the receive channels to generate range-doppler (RD) heatmaps of size 64 x 128, which contain information about the range and velocity of objects. The zero doppler bins in the RD heatmaps are discarded to eliminate static clutter. Subsequently, incoherent addition is performed across doppler bins, followed by a 16-point FFT across the 6 synthetic antenna channels. This process results in the generation of range-angle heatmaps sized 64 x 16. These heatmaps are stored in the L3 memory, which is shared between the processors.

The Cortex®-M4 processor accesses these heatmaps and performs incoherent addition across the range bins. This process generates a doppler column of length 128 and an angle column of length 16. Since the majority of the velocity information is concentrated within ±16 bins from the zero doppler bin, the velocity column is cropped to 32 bins. To match the shape with the VT column, the AT column is expanded to a length of 32 by duplicating its values. By concatenating these columns across 16 frames, VT and AT feature maps are generated, which are then fed to the inference engine for classification.

V. GESTURE RECOGNITION NETWORK: ML-BASED CLASSIFICATION ENGINE

In order to accurately classify the hand gestures captured by tinyRadar, we employed a depthwise separable convolution network as described in Figure 3. The depthwise separable convolution model starts with a 2D convolution filter of size 3 x 3, generating seven output feature maps. Depthwise convolution is then applied to each feature map to capture temporal features, followed by max pooling. This helps reduce the number of parameters. A separable convolution is then

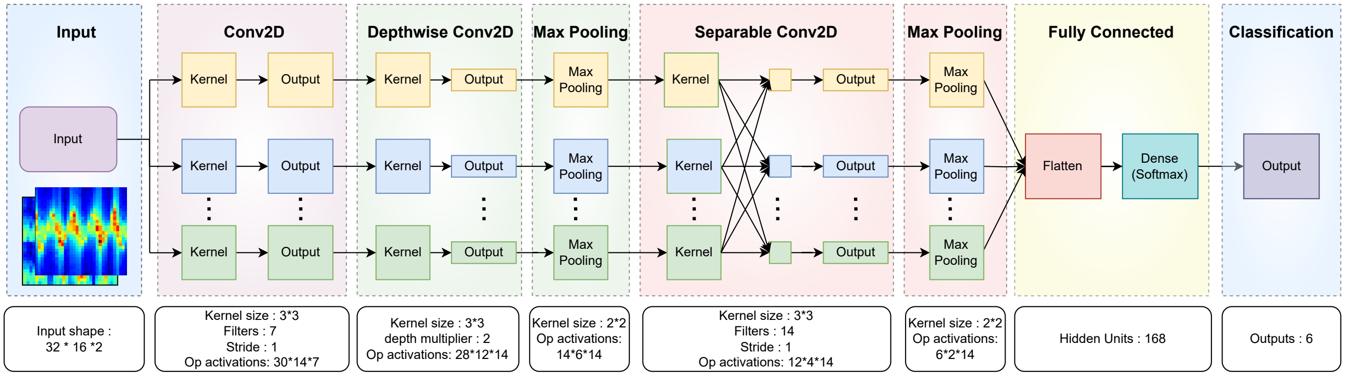

Fig. 3. Gesture recognition architecture based on depthwise separable convolutions

performed, combining depthwise and pointwise convolutions. This approach allows independent learning of each feature map, which are subsequently merged to create an optimized representation. Instead of using a dense layer, the resulting features are directly passed to a softmax classifier for classification, thereby saving memory. This network architecture is designed for efficient computation and parameter usage, making it suitable for real-time inference on resource-constrained devices.

For training the network, we utilized data from eight individuals for training and data from the remaining one individual for testing. The training process involved 200 epochs using the Tensorflow-Lite framework with quantize aware training, where we quantized the network to 8-bit precision. The input feature maps were normalized to signed 8-bit, ensuring consistent representation across the training data. We employed categorical cross-entropy loss with L2 regularization and the Adam optimizer. This combination of techniques improved the network's performance and generalization ability during the training process. To further enhance performance and robustness, we employed an ensemble training approach. This involved training multiple instances of the network while shuffling the users in the training and testing datasets. The predictions of these individual networks were combined by averaging them, resulting in a final network that exhibits improved performance and enhanced robustness.

VI. Results

The radar signal processing and classification network were successfully implemented on the IWRL6432 platform. The signal processing chain utilized $\sim$ 232 KB of memory in the L3RAM, while the classification network required $\sim$ 11 KB of memory on the Cortex®-M4 microcontroller. Classification results were obtained using a sliding window technique with a window size of 4 frames, enabling output generation every 513ms (125ms x 4 frames + 13ms latency). More detailed information about the memory footprint and processing latency for each stage can be found in TABLE II.

When assessed on seven test participants, the quantized inference engine achieved a real-time classification accuracy of 95%. The accuracy per gesture class for recognition on the hardware platform can be observed in TABLE III, which showcases the confusion matrix results. The entire system, comprising the radar frontend, signal processing chain, and classification engine, operated within a power consumption of < 80mW.

TABLE II
Hardware Results

Processing Stage		Core	Memory (KB)	Latency (ms)
Range processing		HWA	192	∼25.6
Doppler & Azimuth processing		HWA	36	∼18
Feature map generation		M4	3.75	∼2.1
Classification network	Parameters	M4	1.55	∼13
	Activation		9.6	

TABLE III
Confusion Matrix in Percentage

Predicted → Actual↓	A	B	C	D	E	F
(A) Circle	93.73	1.14	1	0.71	0.71	2.71
(B) Slow swipe	2.29	91.43	4.56	0.43	1.29	0
(C) Fast swipe	0	1	97.57	0.14	1.29	0
(D) Push	0.29	0.86	0	95.42	2.57	0.86
(E) Cross	0.14	0.29	1.56	0.71	97.3	0
(F) Rest	4.56	0.29	0	0.29	0	94.86

VII. Conclusions

We presented the tinyRadar solution, a low-power, real-time, on-edge radar-based HGR system. The integration of a compact depthwise separable convolution network with the Texas Instruments IWRL6432 radar board enabled accurate and efficient gesture recognition. The system achieved a real-time inference accuracy of 95% for six gestures, with power consumption below 80mW and a total memory footprint of less than 256 KB. tinyRadar framework can be used in smart home automation systems, and interactive IoT devices, and for enhancing accessibility for individuals with limited mobility. We are the first ones, to the best of our knowledge, to develop a low-power ML solution on the IWRL6432 board. We have also developed a working demonstration of our tinyRadar solution for controlling a video interface. To see our solution in action, please visit the following link: https://www.youtube.com/watch?v=RMG7ha1RNHk.

References

[1] L. Chen, F. Wang, H. Deng, and K. Ji, "A survey on hand gesture recognition," in *2013 International conference on computer sciences and applications*. IEEE, 2013, pp. 313–316.

[2] S. Anwar, S. K. Sinha, S. Vivek, and V. Ashank, "Hand gesture recognition: A survey," in *Nanoelectronics, Circuits and Communication Systems: Proceeding of NCCS 2017*. Springer, 2019, pp. 365–371.

[3] A. K. H. Al-Saedi and A. H. H. Al-Asadi, "Survey of hand gesture recognition systems," in *Journal of Physics: Conference Series*, vol. 1294, no. 4. IOP Publishing, 2019, p. 042003.

[4] Y. Murhij and V. Serebrenny, "Hand gestures recognition model for augmented reality robotic applications," in *Proceedings of 15th International Conference on Electromechanics and Robotics" Zavalishin's Readings" ER (ZR) 2020, Ufa, Russia, 15–18 April 2020*. Springer, 2021, pp. 187–196.

[5] A. Khurshid, R. Grunitzki, R. G. Estrada Leyva, F. Marinho, and B. Matthaus Maia Souto Orlando, "Hand gesture recognition for user interaction in augmented reality (ar) experience," in *Virtual, Augmented and Mixed Reality: Design and Development: 14th International Conference, VAMR 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings, Part I*. Springer, 2022, pp. 306–316.

[6] H. Liu, Y. Wang, A. Zhou, H. He, W. Wang, K. Wang, P. Pan, Y. Lu, L. Liu, and H. Ma, "Real-time arm gesture recognition in smart home scenarios via millimeter wave sensing," *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 4, no. 4, pp. 1–28, 2020.

[7] M. Oudah, A. Al-Naji, and J. Chahl, "Hand gesture recognition based on computer vision: a review of techniques," *journal of Imaging*, vol. 6, no. 8, p. 73, 2020.

[8] R. E. Nogales and M. E. Benalcázar, "Hand gesture recognition using machine learning and infrared information: a systematic literature review," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 10, pp. 2859–2886, 2021.

[9] B. K. Chakraborty, D. Sarma, M. K. Bhuyan, and K. F. MacDorman, "Review of constraints on vision-based gesture recognition for human–computer interaction," *IET Computer Vision*, vol. 12, no. 1, pp. 3–15, 2018.

[10] A. Carfì, C. Motolese, B. Bruno, and F. Mastrogiovanni, "Online human gesture recognition using recurrent neural networks and wearable sensors," in *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2018, pp. 188–195.

[11] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *Ieee Access*, vol. 6, pp. 14 410–14 430, 2018.

[12] K. Kudrinko, E. Flavin, X. Zhu, and Q. Li, "Wearable sensor-based sign language recognition: A comprehensive review," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 82–97, 2020.

[13] F. Serpush, M. B. Menhaj, B. Masoumi, and B. Karasfi, "Wearable sensor-based human activity recognition in the smart healthcare system," *Computational intelligence and neuroscience*, vol. 2022, 2022.

[14] Q. Chen, Y. Li, Z. Cui, and Z. Cao, "A hand gesture recognition method for mmwave radar based on angle-range joint temporal feature," in *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2022, pp. 2650–2653.

[15] S. S. Yadav, R. Agarwal, K. Bharath, S. Rao, and C. S. Thakur, "tinyradar: mmwave radar based human activity classification for edge computing," in *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2022, pp. 2414–2417.

[16] S. S. Yadav, R. Agarwal, K. Bharath, S. Rao, and C. Singh Thakur, "tinyradar for fitness: A contactless framework for edge computing," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 17, no. 2, pp. 192–201, 2023.

[17] M. Eggimann, J. Erb, P. Mayer, M. Magno, and L. Benini, "Low power embedded gesture recognition using novel short-range radar sensors," in *2019 IEEE SENSORS*. IEEE, 2019, pp. 1–4.

[18] V. Clarkson and B. Ng, "A radar revolution," *IEEE Potentials*, vol. 38, no. 4, pp. 6–8, 2019.

[19] L. Gan, Y. Liu, Y. Li, R. Zhang, L. Huang, and C. Shi, "Gesture recognition system using 24 ghz fmcw radar sensor realized on real-time edge computing platform," *IEEE Sensors Journal*, vol. 22, no. 9, pp. 8904–8914, 2022.

[20] C. Du, L. Zhang, X. Sun, J. Wang, and J. Sheng, "Enhanced multi-channel feature synthesis for hand gesture recognition based on cnn with a channel and spatial attention mechanism," *IEEE Access*, vol. 8, pp. 144 610–144 620, 2020.

[21] M. Scherer, M. Magno, J. Erb, P. Mayer, M. Eggimann, and L. Benini, "Tinyradarnn: Combining spatial and temporal convolutional neural networks for embedded gesture recognition with short range radars," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 336–10 346, 2021.

[22] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous gesture sensing with millimeter wave radar," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–19, 2016.

[23] 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 *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, 2016, pp. 851–860.

[24] "IWRL6432 Single-Chip 57-to 64-GHz Industrial Radar Sensor," https://www.ti.com/lit/ds/symlink/iwrl6432.pdf, accessed: June 22, 2023.