FPGA Accelerator for Radar-Based Human Activity Recognition

Kangjie Long[1], Chaolin Rao[1], Xiangyu Zhang[1], Wenbin Ye[2], Xin Lou[1]
[1] School of Information Science and Technology, ShanghaiTech University, Shanghai, China
[2] College of Electrical Science and Technology, Shenzhen University, Shenzhen, China
{longk, raochl, zhangxy10, louxin}@shanghaitech.edu.cn, {yewenbin}@szu.edu.cn

***Abstract*—Deep learning is enabling radar-based human activity recognition (HAR) to be used in various application scenarios. In many applications, HAR systems need to be implemented in edge devices with the requirement of real-time processing. This paper presents a high-performance and energy efficient FPGA accelerator for radar-based HAR. The accelerator implements the state-of-the-art Mobile RadarNet algorithm to deliver high precision recognition of human activities. Specific architecture with optimized data is proposed to improve the overall computation efficiency. The accelerator is implemented in a field programmable gate array (FPGA)-based system on chip (SoC) platform. Experimental results show that the overall performance in terms of processing speed and energy efficiency is significantly improved without affecting the recognition accuracy.**

***Index Terms*—human activity recognition (HAR), radar, convolutional neural network (CNN), edge computing**

I. Introduction

In recent years, human activity recognition (HAR) technology has been widely used in various scenarios such as smart homes and autonomous driving [1]. In the past decades, a number of HAR algorithms based on traditional machine learning algorithms such as decision trees [2], support vector machine (SVM) [3] and principal component analysis (PCA) [4], have been proposed. Thereafter, the accuracy of HAR systems had been significantly improved [5] with the help of deep learning. Apart from accuracy, many of these applications demand real-time processing, and in the meantime, most of them needs to be performed in edge devices without powerful computing devices such as graphic processing units (GPUs) [6]. However, deep learning models tend to be much more complex to achieve higher accuracy of recognition, such that a large amount of computing and memory resources are needed. Therefore, specifically designed accelerators are necessary for the deployment of HAR systems in real-life applications.

The mainstream HAR systems can be divided into two categories: vision-based and radar-based [7]. Although vision-based technologies are mature and easy to deploy, they have some critical drawbacks for HAR. Firstly, data collection via vision-based technology has the risk of personal privacy disclosure, and therefore is not possible to be deployed in private spaces. Secondly, quality of illumination and other conditions such as weather can greatly affect the effectiveness of optical imaging [8]. On the other hand, radar-based HAR is illumination-invariant, and will not cause privacy problems. For a radar-based HAR system, the most critical task is to extract features from the raw radar data. So far, there are many methods to extract features from the radar spectrum, such as PCA and dynamic time warping [9], [10]. These methods either rely heavily on manual operations or difficult to generalize. Learning-based feature extraction does not demand extensive prior knowledge, and is much more robust [11]. Therefore, deep learning are becoming popular for radar-based HAR system design. However, unlike vision-based approaches, where there exist many dedicated hardware designs in literature, hardware accelerator for radar-based HAR system is seldom discussed. Moreover, due to the uniqueness of radar signals, it is necessary to develop specific hardware accelerators for radar-based HAR, instead of directly using the existing designs for vision-based applications.

In this paper, a dedicated hardware accelerator design based on Mobile RadarNet, a state-of-the-art HAR algorithm, is proposed. A specific hardware architecture is designed to efficiently explore the advantage of Mobile RadarNet, where processing elements (PEs) with soft-coupled data flow are developed to improve the overall computation efficiency of the system. To demonstrate the proposed system, it is mapped to a field programmable gate array (FPGA)-based system on chip (SoC) platform. Experimental results show that the processing speed and energy efficiency are significantly improved compared to the original software implementation, while the recognition accuracy is maintained as high as the original design.

II. Background

In this Section, the background of Doppler radar signal is explained. The state-of-the-art Mobile RadarNet [12], which can be used to recognize the seven human activities is also briefly introduced.

A. Doppler Radar Signal

When vibrations or rotations of targets have been detected, the frequency of the radar echoes induces additional frequency modulations. This phenomena is always referred to as micro-Doppler [13]. Accordingly, the radar echoes can be regarded as a superposition of multiple frequency components, which reflects the velocity of vibration from different parts of the detected targets. Therefore, distinct features radar echoes will be resulted in from different human activities. Since Doppler radar echo is a time series, it is difficult to extract features

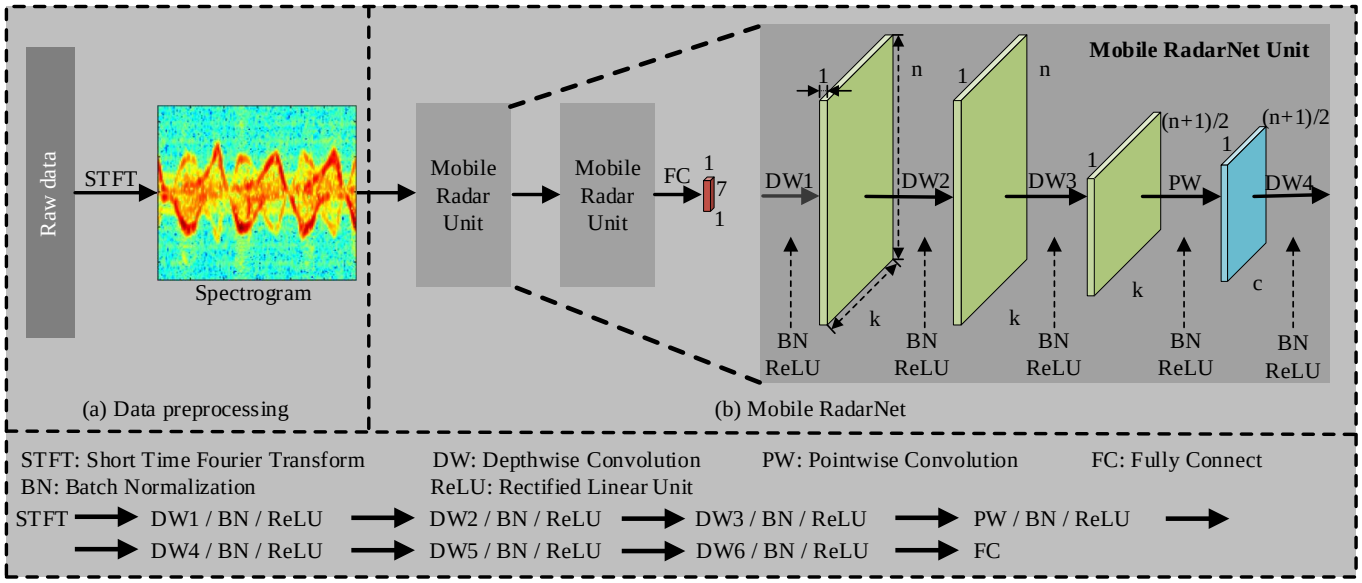

Fig. 1. Architecture of Mobile RadarNet [12].

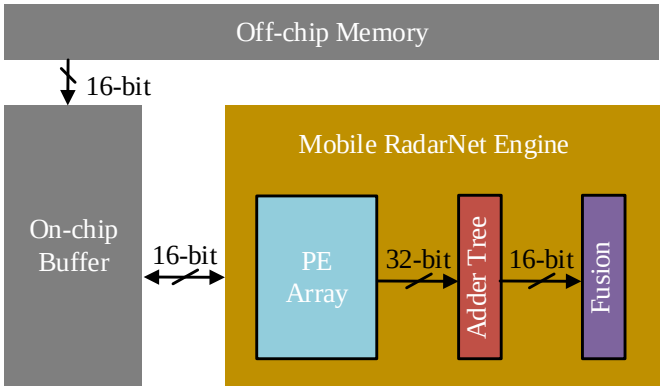

Fig. 2. Overall hardware architecture of the proposed Mobile RadarNet accelerator.

directly because Doppler signals are easily contaminated. A common practice is to transform time series into frequency domain [14]. In Mobile RadarNet [12], a Short-Time Fourier Transform (STFT) is performed on the signal of the radar echo to obtain a spectrogram with frequency-time joint variables. It is worth noting that the Doppler frequency of radar echo corresponds to the speed of moving targets, where the movements of the detected targets are all coherent and do not change drastically within a short period of time.

B. Mobile RadarNet

The Mobile RadarNet [12] is a state-of-the-art algorithm for HAR. As shown in Fig. 1, it consists of two Mobile RadarNet Units and a fully connected layer, where the Mobile RadarNet unit is composed of three depth-wise convolutions and one point-wise convolution. The correlation of different channels is not cohesive when the frequency of spectrogram is treated as the channel of the input feature map. Therefore, there is no need for each point-wise convolution to be connected to a depth-wise convolution for message exchanging. Furthermore, the fully connected convolution has the same ability to exchange messages of different channels. Thus, the point-wise convolution of the second Mobile Radar Unit is replaced by a fully connected layer. And then, each depth-wise separable convolution and point-wise separable convolution layer is followed by a batch normalization layer and a ReLU layer. In data pre-processing of [12], a 150-points STFT is performed on the original human activity time series, where the length of Hamming window is $25.5ms$ and the overlap is 12 sampling points. Each human activity time series after STFT results in a time-frequency joint variable spectrogram with time pixel and frequency pixel of 153 and 150, respectively.

III. Proposed Hardware Architecture

In this section, a dedicated hardware architecture for accelerating the Mobile RadarNet is proposed. Shown in Fig. 2 is the overall architecture of the proposed accelerator, which consists of a Mobile RadarNet Engine and a couple of storing modules. Software-hardware co-design technique is applied in this work to deal with the interaction of data between off-chip memory and on-chip buffer. With the aim of efficient implementation, the idea of block-circulant computing is applied in the design of Mobile RadarNet Engine. In order to maintain the accuracy of recognition while reducing the computational complexity, linear dynamic quantization is introduced in this work to quantize the weight of the model.

A. The Mobile RadarNet Engine

The details of the proposed Mobile RadarNet accelerator is illustrated in Fig. 3, where the Mobile Radar Engine is the most critical part. As illustrated in Fig. 3 (a), the processing element (PE) array consists of 4×4 PEs, with each PE contains one multiplier and one register. The adder tree shown in Fig. 3 (b) is used to sum up the output of PE Array to generate the output feature maps. The Fusion module in Fig. 3 (b) is used to implement the normalization operations in the engine. In the Fusion module, the bias of convolution is fused into the operation of normalization to reduce memory access and

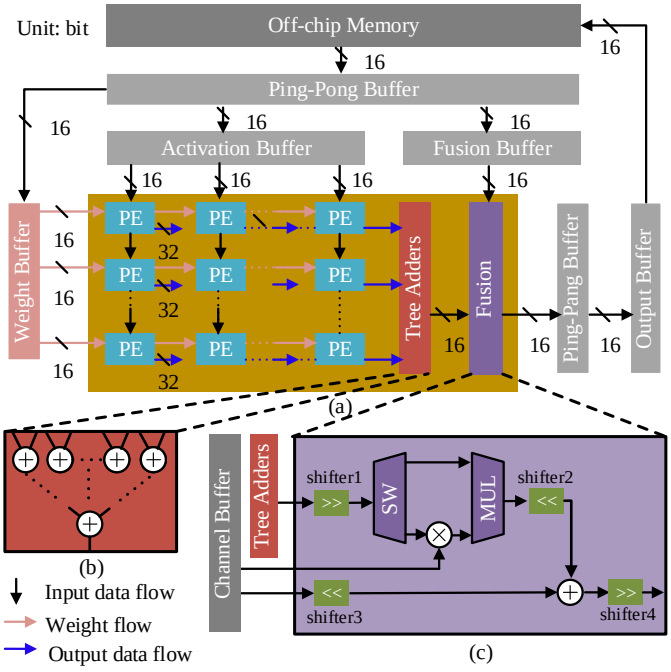

Fig. 3. Detailed architecture of the proposed Mobile RadarNet accelerator.

computational complexity. Moreover, the Fusion module uses a switch (denoted by SW) and a multiplexer (denoted by MUL) to bypass the batch normalization for the computation of fully connected layer. Apart from the Mobile RadarNet Engine, a couple of on-chip buffer is used to improve the computation efficiency of the Mobile RadarNet Engine. For example, the Ping-Pang Buffer can help to hide the latency of off-chip memory access.

In this work, a linear dynamic fixed-point quantization technique based on data statistics is applied. In the fixed-point number system, numbers are represented by (s, n, m), where s, n and m indicate the number of bits for sign, integer part and fractional part, respectively. Based on the data statistics, n and m may be different for each layer to represent the input features, weights, biases, etc. Note that n and m can be obtained for each layer during the training process. Moreover, as shown in Fig. 3 (c), shifter1 can be used to adjust the bits of fraction of the output of adder tree to accord with m. The shifter2 and shifter3 are used to ensure that the m of parameters of MUL and Channel Buffer are the same before adding. The shifter4 is used to adjust the m for the next layer.

B. Data Flow

To implement Mobile RadarNet, the PE Array need to support not only depth-wise convolution but also point-wise convolution. Therefore, the corresponding data flow needs to be specifically designed to maximize the reuse of data for these two different convolution operations. An example of point-wise convolution is illustrated in Fig. 4 (a), where the size of PE Array is 3×3 for the convinience of explanation. In cycle=0, three pixels of weight (denoted by PofW) and nine pixels of input feature maps (denoted by PofIFM) are loaded to the PE Array for multiplication. In cycle=1, three PofWs and three out of the nine PofIFMs are updated to the PE Array

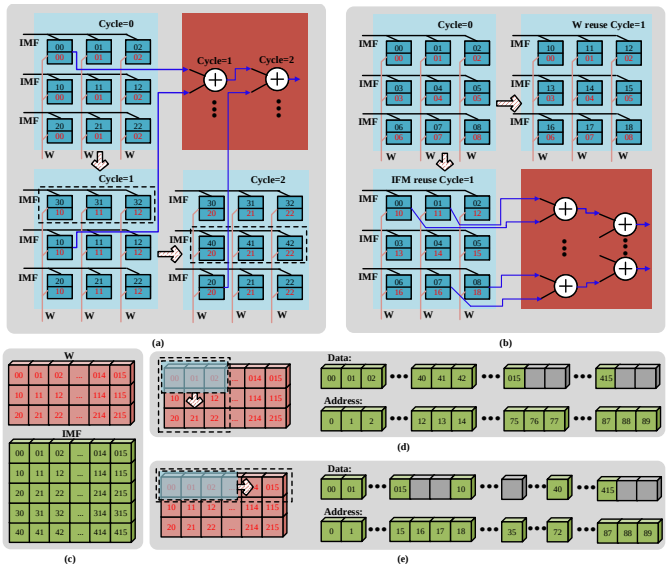

Fig. 4. Illustration of data flow.

to complete a similar computation as in cycle=0. Therefore, 66.7% PofIFMs can be reused in cycle=1. Accordingly, the data of Fig. 4 (c) needs to be rearranged in memory as depicted in Fig. 4 (d) to adapted to the data flow, where the Data represents the arrangement of input feature map (IFM) and the gray parts are padded with zeros to ensure the alignment for each computation. Similarly, an example of point-wise convolution is depicted in Fig. 4 (b), which enables the reuse of PofW and PofIFM. The data flow of PofW and PofIFM are the same in cycle=0, where nine PofWs and PofIFMs are loaded to the PE Array for multiplication. In cycle=1, nine PofWs are updated from the next channel, while nine PofIFMs are maintained the same. Contrarily, nine PofIFMs are updated from the next channel, while nine PofWs are maintained the same in cycle=1 for the reuse of PofW. Therefore, the amount of memory access for PofIFM and PofW can be given by

$$M_W = C_{in} \times H \times C_{out} + C_{out} \times C_{in} \tag{1}$$

and

$$M_{IFM} = C_{in} \times H + H \times C_{out} \times C_{in} \tag{2}$$

respectively, where C_{in}, C_{out} and H represent the number of channels of input feature, the number of filters and the height of output feature map, respectively. Finally, the computation flow can be determined by

$$\frac{M_W}{M_{IFM}} = \frac{C_{out} \times H + C_{out}}{C_{out} \times H + H} \tag{3}$$

An example of arrangement of IFM is illustrated in Fig. 4 (e) for the data flow of point-wise convolution. Among them, the gray parts are padded with zeros to ensure the height (18 after padding) of IFM is divisible by the amount (9 in this example) of PE.

IV. FPGA Implementation Results

In this section, we present the implementation results of the FPGA-based Mobile RadarNet accelerator. To demonstrate

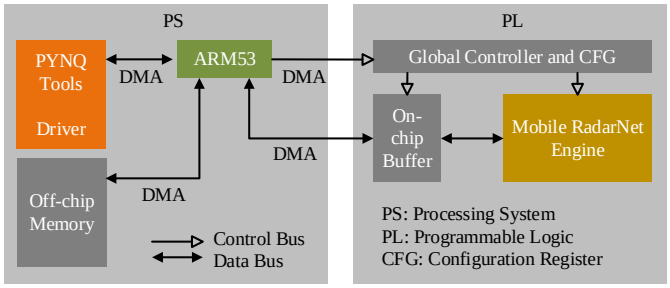

Fig. 5. Overview of the accelerator implementation in the SoC platform.

the effectiveness of the proposed design, we compared the proposed accelerator with CPU-based, GPU-based and another related FPGA-based HAR accelerators.

In this work, the Mobile RadarNet accelerator is implemented on a Xilinx Zynq UltraScale+ ZCU102 evaluation platform. The overall architecture of the proposed accelerator system is depicted in Fig. 5, which can be divided into two parts, the processing system (PS) part and the programmable logic (PL) part. The Mobile RadarNet Engine is implemented using the PL, which is the most critical part of the proposed accelerator. The on-chip Buffer is used to cache the interactive data between Mobile RadarNet Engine in the PL part and off-chip memory in the PS part. The ARM53 in the PS part is the master controller, which is responsible for the management of off-chip DDR4 memory access and other system controls. It is also used for the configuration of the registers via AXI4-lite, which in turn controls the computation and memory access of Mobile RadarNet Engine. Note that on-chip Buffer refers to a collection of buffers, including Ping-Pang Buffer, Activation Buffer, Weight Buffer, Fusion Buffer and Output Buffer in Fig. 3.

Table I lists the FPGA implementation results of the proposed accelerator as well as the results of CPU, GPU and other FPGA-based HAR designs. As we can see, running at a clock frequency of 150 MHz, the power of proposed accelerator is 24W, which is lower than the existing design [15]. Specifically, there is no existing radar-based FPGA accelerator for HAR, and the design in [15] is an accelerator based on 3-D CNN for vision-based HAR, which we find is the most related design to the proposed. The Energy per multiply and add (MAC) of the proposed design is less than 10% of that in [15]. Moreover, to show the baselines, we also present the implementation results of Mobile RadarNet on CPU and GPU. As shown in Table I, the Eenergy per MAC of the proposed design is significantly reduced without affecting final recognition accuracy.

TABLE I
PERFORMANCE COMPARISON.

	CPU	GPU	FPGA [15]	**This Work**
Platform	Intel i7-7700	GeForce GTX 1080 Ti	Intel Arria 10 SX660	**Zynq XCZU9EG**
Frequency	3.6GHz	1.48GHz	150MHz	**150MHz**
Process	14nm	16nm	20nm	**16nm**
Model	Mobile RadarNet	Mobile RadarNet	E3DNet	**Mobile RadarNet**
No. of batches	1	64	1	**1**
Accuracy	97.3%	97.3%	85.2%	**97.3%**
Power (W)	140	208	36	**24**
MAC per frame	0.61G	0.61G	6.10G	**0.61G**
Energy per MAC (J/G)	159.381	0.928	0.201	**0.015**

V. CONCLUSION

This paper presents a high-performance and energy efficient FPGA accelerator for radar-based HAR. The accelerator implements the state-of-the-art Mobile RadarNet algorithm to deliver high precision recognition of human activities. Data flow as well as quantization are specifically designed to take advantage of Mobile RadarNet. The accelerator is implemented in Xilinx ZCU102, an FPGA-based SoC platform. Experimental results show that, without sacrificing the recognition accuracy, the processing speed and energy efficiency is significantly improved.

REFERENCES

[1] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *Pervasive Computing, IEEE*, vol. 9, no. 1, pp. P.48–53, 2010.
[2] A. Reiss and D. Stricker, "Personalized mobile physical activity recognition," *ACM*, p. 25, 2013.
[3] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyesortiz, "A public domain dataset for human activity recognition using smartphones," 2013.
[4] W. Li, B. Xiong, and G. Kuang, "Target classification and recognition based on micro-doppler radar signatures," in *2017 Progress in Electromagnetics Research Symposium-Fall (PIERS-FALL)*. IEEE, 2017, pp. 1679–1684.
[5] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, pp. 1–33, 2014.
[6] R. Brito, R. P. Biuk-Aghai, and S. Fong, "Gpu-based parallel shadow features generation at neural system for improving gait human activity recognition," *Multimedia Tools and Applications*, vol. 80, no. 8, pp. 12 293–12 308, 2021.
[7] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.
[8] E. B. Gindele, "Method of compensating a digital image for the effects of flare light," *US*, 2003.
[9] Y. Kim and H. Ling, "Human activity classification based on micro-doppler signatures using a support vector machine," *IEEE transactions on geoscience and remote sensing*, vol. 47, no. 5, pp. 1328–1337, 2009.
[10] Z. Zhou, Z. Cao, and Y. Pi, "Dynamic gesture recognition with a terahertz radar based on range profile sequences and doppler signatures," *Sensors*, vol. 18, no. 1, p. 10, 2018.
[11] Y. Kim and T. Moon, "Human detection and activity classification based on micro-doppler signatures using deep convolutional neural networks," *IEEE geoscience and remote sensing letters*, vol. 13, no. 1, pp. 8–12, 2015.
[12] J. Zhu, X. Lou, and W. Ye, "Lightweight deep learning model in mobile-edge computing for radar-based human activity recognition," *IEEE Internet of Things Journal*, vol. 8, no. 15, pp. 12 350–12 359, 2021.
[13] Y. Yang, J. Lei, W. Zhang, and L. Chao, "Target classification and pattern recognition using micro-doppler radar signatures," in *Acis International Conference on Software Engineering*, 2006.
[14] V. C. Chen and R. D. Lipps, "Time frequency signatures of micro-doppler phenomenon for feature extraction," in *Wavelet Applications VII*, vol. 4056. International Society for Optics and Photonics, 2000, pp. 220–226.
[15] H. Fan, C. Luo, C. Zeng, M. Ferianc, Z. Que, S. Liu, X. Niu, and W. Luk, "F-e3d: Fpga-based acceleration of an efficient 3d convolutional neural network for human action recognition," in *2019 IEEE 30th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, vol. 2160. IEEE, 2019, pp. 1–8.