Received May 25, 2019, accepted June 21, 2019, date of publication July 2, 2019, date of current version July 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926381

Acceleration of FPGA Based Convolutional Neural Network for Human Activity Classification Using Millimeter-Wave Radar

PENG LEI[1], JIAWEI LIANG[1], ZHENYU GUAN[2], (Member, IEEE), JUN WANG[1], AND TONG ZHENG[1]

[1]School of Electronic and Information Engineering, Beihang University, Beijing 100191, China
[2]School of Cyber Science and Technology, Beihang University, Beijing 100191, China

Corresponding author: Zhenyu Guan (guanzhenyu@buaa.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61501011 and Grant 61671035.

ABSTRACT Deep learning techniques have attracted much attention in the radar automatic target recognition. In this paper, we investigate an acceleration method of the convolutional neural network (CNN) on the field-programmable gate array (FPGA) for the embedded application of the millimeter-wave (mmW) radar-based human activity classification. Considering the micro-Doppler effect caused by a person's body movements, the spectrogram of mmW radar echoes is adopted as the CNN input. After that, according to the CNN architecture and the properties of the FPGA implementation, several parallel processing strategies are designed as well as data quantization and optimization of classification decision to accelerate the CNN execution. Finally, comparative experiments and discussions are carried out based on a measured dataset of nine individuals with four different actions by using a 77-GHz mmW radar. The results show that the proposed method not only maintains the high classification accuracy but also improves its execution speed, memory requirement, and power consumption. Specifically, compared with the implementation of the same network model on a graphics processing unit, it could achieve the speedup of about 30.42% at the cost of the classification accuracy loss of only 0.27%.

INDEX TERMS Human activity classification, millimeter-wave radar, convolutional neural network, field-programmable gate array, acceleration.

I. INTRODUCTION

Millimeter wave (mmW) radars are an emerging and promising technique in sensing systems. They generally operate from about 30 GHz to 300 GHz [1]. Although severe atmospheric attenuation limits their usefulness involved in the long-range propagation, the millimeter-level waves of signals provide some noticeable advantage in relatively short-range sensing systems on the Earth. First, small components of mmW radars are beneficial for the miniaturization of equipment. For example, since the antenna's physical aperture is approximately proportional to its gain and square of wavelength, mmW radars allow a small-size yet high-gain antenna. Second, large bandwidth, e.g., 4 GHz [2], is available in mmW radars. It helps enhance their resolution and accuracy performance in the range measurement. Third, the short wavelength means the high Doppler sensitivity, which may improve their capabilities in targets' macro- and micro-motion analysis. Furthermore, the fast development of compound semiconductor technology in recent years advances and affords the implementation of mmW radar systems on chip. They have attracted much attention in many civilian applications, such as non-contact controlling devices [3], driver assistant systems [4]–[7], foreign objects debris detection [8], and so on.

The automatic target recognition is a desired capability in modern radars. It is well-known that the human activity classification is one of hot topics because of its wide involvement in a variety of radar applications [9]–[11]. Chen *et al.* introduce the radar micro-Doppler (mD) effect in the human identification [12], which represents the frequency modulation in radar echoes caused by micro-motions of targets, e.g., human gait, wheel rotation, and so forth. On the basis, a variety of radar mD features are extracted for the classification of

The associate editor coordinating the review of this manuscript and approving it for publication was Jianhua He.

different human activities and the estimation of total energy expenditure [13], [14]. In [15], a Bayesian classifier is given to distinguish between humans and vehicles by using radar mD signals. With the success of deep learning techniques in computer vision, they have become much popular in the radar mD based classification of human targets over past few years. Such methods are able to automatically find useful features instead of the handcrafted feature extraction in traditional approaches. A deep convolutional neural network (CNN) is presented in [16] for both human detection and activity classification. In [17], a Bayesian learning method is employed to optimize the CNN architecture in the human activity classification with mD radar signals. The fusion of time-frequency distribution and range map of radar mD signals is used as inputs of stacked autoencoders for the fall detection [18].

It should be noted that most of previous deep learning based studies rely on the graphics processing unit (GPU) hardware. However, for their implementation in embedded systems, field programmable gate arrays (FPGAs) are the favorable platform nowadays owing to the small size, low power consumption, parallel processing, public availability, and so forth. Different from GPU, the optimization of computation and memory resource usage as well as fixed point arithmetic operations with restricted bitwidth would be important issues in the acceleration of deep learning models on FPGAs. In [19], a CNN accelerator on FPGA is designed for the ImageNet classification. The dynamic-precision data quantization and the singular value decomposition are used to reduce computation burden and memory footprint, respectively. A systematic design space exploration method is provided in [20] to maximize the throughput of an OpenCL based FPGA accelerator for CNN models. In [21], a CNN architecture is designed for the latency-centric optimization on FPGA by using the weights reloading transformation. In this paper, considering the characterization of time-frequency representation of human mD signals, a FPGA based CNN acceleration method is proposed for the human activity classification by using mmW mD signals. Different from previous work, it explores both the computation procedure of CNN and the pipelining technique on FPGA to improve the execution speed in the forward propagation through the network. In particular, several parallel processing schemes are provided for the computation implementation involved in channels and layers of the CNN, respectively. They help improve the speed of the network on FPGA by reducing the waiting time and caching time.

This paper is organized as follows. Section II briefly introduces the time-frequency representation of human mD signals in the mmW radar with frequency-modulated continuous wave (FMCW) waveforms. Section III describes the CNN architecture with forward-propagation procedure. The acceleration method of CNN based the human activity classification on FPGA is presented in Section IV. Experiments and result discussion using measured data at 77 GHz are carried out to validate the proposed method in Section V. Section VI gives the conclusion.

II. MICRO-DOPPLER SPECTROGRAM IN FMCW MMW RADAR

Radar mD features characterize the time-varying frequency modulation of target's micro-kinematics in received echoes. In addition, due to their uniqueness for different targets with micro-motions [22], the time-frequency representation of mD signals could offer some advantage in the human activity classification. Especially in the case of mmW radars, their short wavelength, which may cause wide dynamic range of mD frequency corresponding to human gait patterns, gives the benefit of mD measurement in the time-frequency domain.

Let us consider the linear FMCW waveform with sawtooth-shaped frequency sweep, which is referred to as FMCW for simplification if not especially noted in the paper. The transmitted FMCW signal can be expressed as

$$x(\tau) = rect\left(\frac{\tau}{T}\right) \cdot \exp\left[j2\pi\left(f_c\tau + \rho\frac{\tau^2}{2}\right) + j\varphi_0\right] \tag{1}$$

where τ is the fast time, T is the sweep period, f_c is the carrier frequency, ρ is the frequency sweep rate, φ_0 is the initial phase, and $rect(\cdot)$ is the unit rectangular window function with the width of 1. Assume that the instantaneous distance of any point scatterer on the human body from the radar is $R(t)$, where t is the slow time. Then the received FMCW signal is

$$\begin{aligned} y(\tau, t) = A \cdot rect&\left(\frac{\tau - 2R(t)/c}{T}\right) \\ \cdot \exp&\left\{j2\pi\left[f_c\left(\tau - \frac{2R(t)}{c}\right)\right.\right. \\ &\left.\left. + \rho\frac{\left(\tau - 2R(t)/c\right)^2}{2}\right] + j\varphi_0\right\} \end{aligned} \tag{2}$$

where A is the amplitude, and c is the speed of light. The amplitude fluctuation is ignored herein to simplify the analysis.

Using the stretch processing [23], we can obtain the beat signal as

$$\begin{aligned} &y_{beat}(\tau, t) \\ &= A \cdot rect\left(\frac{\tau - 2R(t)/c}{T - 2R(t)/c}\right) \cdot \exp\left\{j2\pi\left[\frac{2\rho R(t)}{c}\tau\right.\right. \\ &\quad\left.\left. + \frac{2f_cR(t)}{c} - \frac{2\rho R^2(t)}{c^2}\right]\right\} \\ &\quad + rect\left(\frac{\tau - T}{2R(t)/c}\right) \cdot \exp\left\{j2\pi\left[\left(\frac{2\rho R(t)}{c} - B_W\right)\tau\right.\right. \\ &\quad\left.\left. + \frac{2f_cR(t)}{c} - \frac{2\rho R^2(t)}{c^2}\right] + j\theta_0\right\} \end{aligned} \tag{3}$$

where $B_W = \rho T$ is the bandwidth, and the constant phase term θ_0 is given by

$$\theta_0 = \pi B_W T - 2\pi f_c T \tag{4}$$

It should be noted that (3) is time divided into two parts according to different intervals of the fast time in unit rectangular window functions. On the basis, we take the fast

Fourier transform (FFT) of signal samples in each of sweeps with respect to τ successively, and the result is denoted by $Y_{beat}(f_\tau, t)$. Since the mmW radar range for human target is generally no more than hundreds of meters, the quadratic term and the second part in (3) could be omitted in $Y_{beat}(f_\tau, t)$. It can be found that for a given slow time t_0, the maximum amplitude of $|Y_{beat}(f_\tau, t_0)|$ appears at

$$f_{\tau\_peak}(t_0) = \frac{2\rho R(t_0)}{c} \tag{5}$$

It indicates that the target distance at the slow time t is estimated as $cf_{\tau\_peak}\big/(2\rho)$.

When there is no human body's translational motion or after compensation for it, we could obtain the baseband signal within the given range gates in $Y_{beat}(f_\tau, t)$ as

$$\begin{aligned} y_{ibase}(t) = A_i \cdot \left[T - 2\frac{R_0 + R_{imd}(t)}{c}\right] \\ \cdot \exp\left\{j2\pi\left[2f_c\frac{R_0 + R_{imd}(t)}{c}\right.\right. \\ \left.\left. -2\frac{\rho\,[R_0 + R_{imd}(t)]^2}{c^2}\right]\right\} \end{aligned} \tag{6}$$

where R_0 and $R_{imd}(t)$ are radial distances of the human's geometric center from the radar and his i-th body part, respectively, and the amplitude A_i is related to the radar cross section of the body part. For the human gait measurement by using mmW radars, $T \gg 2[R_0 + R_{imd}(t)]\big/c$ always holds. Therefore, the baseband echoes from the complete human body micro-motion could be approximated as

$$y_{base}(t) \approx T\sum_{i=1}^{K} A_i \exp\left\{j\left[\frac{4\pi f_c R_{imd}(t)}{c} + \varphi_0\right]\right\} \tag{7}$$

where K denotes the number of human body parts in some corresponding kinematic models, and $\varphi_0 = 4\pi f_c R_0\big/c$. Moreover, different from the radial distance, the instantaneous Doppler shift is proportional to the carrier frequency besides its dependence on the aspect angle of human moving direction from the radar line of sight (LOS). It indicates that the high carrier frequency of mmW radars may mitigate the influence of aspect angles close to 90° to some extent. Therefore, we shall take advantage of high Doppler sensitivity in mmW radars for the human activity classification. Using the short-time Fourier transform (STFT), the spectrogram of baseband echoes could be given by [24]

$$\left|TF\left(t', f\right)\right| = \left|\int_{-\infty}^{+\infty} y_{base}(t)\, w\left(t - t'\right) e^{-j2\pi ft} dt\right| \tag{8}$$

where $w(t)$ denote the window function. Equation (8) actually presents a mapping process of mD signals from the one-dimensional time domain to the two-dimensional time-frequency domain by means of a sliding time window. Since $\left|TF(t', f)\right|$ indicates the time-varying mD frequency modulation caused by human gaits, the spectrogram is used as input to the CNN for human activity classification.

III. CNN ARCHITECTURE AND ITS COMPUTATIONAL WORKLOAD

The CNN is one of the most outstanding deep learning models in the object recognition. It is structured as a pipeline of layers. From the network architecture viewpoint, the CNN applied in the classification generally consists of convolutional layer (Conv), pooling layer, activation function, fully connected layer (FC), and classifier. Some linear and non-linear processing involved in successive groups of these layers is performed for mapping the input spectrogram to a one-dimensional feature space. On the basis, a classifier is used to make the prediction of its category. It should be noted that there are two directions of information flow in CNN, namely, backpropagation in training and forward propagation in test. Since the proposed acceleration method of FPGA based CNN focus on the test phase, we briefly introduce the computation workload theoretically involved in its forward propagation as follows.

A convolutional layer always contains a few two- or three-dimensional kernels for the convolution operation with corresponding input. In this study, the classical square kernel is utilized herein. Let sizes of input data and kernels of the j-th convolutional layer be $M_j \times N_j \times L_j$ and $K_j \times K_j \times L_j$, respectively. Especially for the first convolutional layer, $L_j = 1$ depending on the data dimension of mD spectrogram. Hence, the output of the convolutional layer in the case of stride 1 can be given by

$$\begin{aligned} \mathbf{D}_j^{\mathrm{Conv}O}(m, n, h) = \sum_{l=1}^{L_j}\sum_{q=1}^{K_j}\sum_{p=1}^{K_j} \mathbf{w}_{jh}(p, q, l) \times \mathbf{D}_j^{\mathrm{Conv}I} \\ \times (m + p, n + q, l) + b_{jh}^{\mathrm{Conv}} \end{aligned} \tag{9}$$

where $\mathbf{w}_{jh}$ denotes the h-th kernel of the j-th convolutional layer, $\mathbf{D}_j^{\mathrm{Conv}I}$ and $\mathbf{D}_j^{\mathrm{Conv}O}$ are its input and output data, and b_{jh}^{Conv} is the bias parameter. Assume that there exist H_j kernels in the j-th convolutional layer, namely that the number of channels in this layer is H_j. Considering the distribution of mD signals in spectrogram and feature maps as well as computation efficiency, no zero padding is used herein. Therefore, the dimensionality of output $\mathbf{D}_j^{\mathrm{Conv}O}$ of the j-th convolutional layer is $(M_j - K_j + 1) \times (N_j - K_j + 1) \times H_j$. Accordingly, the amount of multiplication and addition required in the j-th convolutional layer is

$$\begin{aligned} O(\mathrm{Conv}) = 2K_j \times K_j \times L_j \times \left(M_j - K_j + 1\right) \\ \times \left(N_j - K_j + 1\right) \times H_j \end{aligned} \tag{10}$$

An activation function may define a nonlinear mapping from its input to output. The widely-used rectified linear unit (ReLU) [25] and Hard Tanh [26] are adopted herein after convolutional layers and the first fully connected layer, respectively, because of its two merits: 1) the mitigation of gradient vanishing problem in training [27]; 2) computation efficiency. Suppose that the dimensionality of input to the activation function is U. According to properties of these two activation functions, it only involves comparison operations in the amount of

$$\begin{cases} O\left(\text{ReLU}\right) = 1 \times U \\ O\left(\text{HT}\right) = 2 \times U \end{cases} \tag{11}$$

A pooling layer performs the downsampling to ReLU results. It helps reduce the size of feature map to alleviate computational burden for the following layers, and increase the robustness of feature extraction against its translation to some extent. As to the three-dimensional data after the multi-channel convolution and ReLU, i.e., $\mathbf{D}_j^{\text{ReLU}O} = \text{ReLU}(\mathbf{D}_j^{\text{Conv}O})$, the pooling operation is employed to the ReLU result of every submatrix from each of channels in the j-th convolutional layer. In this work, we use the 2 × 2 max pooling, which could be expressed as

$$\begin{aligned} \mathbf{D}_j^{\text{PL}O}\left(m,n,h\right) = \max\Big[&\mathbf{D}_j^{\text{ReLU}O}\left(2m-1,2n-1,h\right), \\ &\mathbf{D}_j^{\text{ReLU}O}\left(2m,2n-1,h\right), \\ &\mathbf{D}_j^{\text{ReLU}O}\left(2m-1,2n,h\right), \\ &\mathbf{D}_j^{\text{ReLU}O}\left(2m,2n,h\right)\Big] \end{aligned} \tag{12}$$

We can see that the max pooling layer require some comparison operations, and their number is

$$O\left(\text{PL}\right) = 3 \times \left\lfloor \frac{M_j - K_j + 1}{2} \right\rfloor \times \left\lfloor \frac{N_j - K_j + 1}{2} \right\rfloor \times H_j \tag{13}$$

where $\lfloor \cdot \rfloor$ denotes the floor function.

A fully connected layer actually makes the matrix multiplication as well as the addition of bias. Denote the FC input vector by $\mathbf{D}^{\text{FC}I}$. The equation of this layer is given by

$$\mathbf{D}^{\text{FC}O} = \mathbf{v}\mathbf{D}^{\text{FC}I} + \mathbf{b}^{\text{FC}} \tag{14}$$

where $\mathbf{D}^{\text{FC}O}$ is the FC output vector, $\mathbf{v}$ is the weight matrix in this fully connected layer, and $\mathbf{b}^{\text{FC}}$ is the corresponding bias parameter vector. Assume that dimensionalities of $\mathbf{D}^{\text{FC}I}$ and $\mathbf{D}^{\text{FC}O}$ are $U^{\text{FC}I}$ and $U^{\text{FC}O}$, respectively. Then the amount of multiplication and addition required in the fully connected layer is

$$O\left(\text{FC}\right) = 2 \times U^{\text{FC}I} \times U^{\text{FC}O} \tag{15}$$

The softmax is a normalized exponential function defined by [28]

$$\eta_k = \exp\left[\mathbf{D}^{\text{FC}O}\left(k,1\right)\right] \Big/ \sum_{j=1}^{U^{\text{FC}O}} \exp\left[\mathbf{D}^{\text{FC}O}\left(j,1\right)\right] \tag{16}$$

It is used herein as a classifier, and converts output values of the last fully connected layer to the form of probabilities corresponding to each class of human activities.

IV. OPTIMIZED IMPLEMENTATION OF CNN ON FPGA

A. QUANTIZATION

Nowadays the development of FPGA technique has allowed more digital signal processor (DSP) slices available on chips. However, many of them still work in the fix-point mode, and computation resources as well as on-chip memory are still limited. It may result in the low efficiency or even incapability of executing traditional CNNs on the FPGA system. Therefore, considering the bitwidth constraint of FPGA, we apply a series of quantization strategies to different types of data used in the CNN model. The 16-bit fixed-point number representation is taken herein for feature maps and weight parameters according to a reasonable compromise between precision and hardware efficiency [29].

First, the spectrogram is normalized to the interval [0, 1]. Due to the non-negativity of its values, quantized results of the input data of CNN on FPGA can be represented by 1 integer bit and 15 fraction bits as

$$\mathbf{D}_{q\_TF} = \left\lfloor \mathbf{D}_{n\_TF} \times 2^{15} \right\rfloor \tag{17}$$

where $\mathbf{D}_{n\_TF}$ is the normalized spectrogram of radar mD signals.

Second, it is found that weight values of the network model yield $-1 < \alpha_{wt} < 1$. Thus we take 1 bit for the sign and 15 bits for the fractional part in the quantization of weights, namely,

$$\alpha_{q\_wt} = \left\lfloor \alpha_{wt} \times 2^{15} \right\rfloor \tag{18}$$

Third, considering the variation of inter-layer data, the dynamic fixed-point quantization is used for feature maps. As stated in Section III, these inter-layer data undergo multiplication and addition in the last convolutional layer as well as the comparison involved in the last activation function and pooling layer. We know that the multiplication may change the bitwidth of fractional part in the data, the addition may change the range of data values, and the activation function could have impact on both. Moreover, when the bitwidth of data is larger than 16 bits, they are truncated according to maximum of cell values in the corresponding feature maps. This guarantees the preservation of all the integer bits in the data.

Last but not least, biases in convolutional and fully connected layers are employed in the addition operation. As shown in (9) and (14), the integer and fraction bits of bias values should be aligned with those of multiplier output, respectively. Therefore, biases are quantized with dynamic configuration.

B. PARALLELIZATION PROCESSING

As a logic device, one advantage of FPGA is the ability of highly parallel computing. In order to explore it for the FPGA based CNN accelerator in human activity classification, four parallel processing strategies are offered as follows according to data flow and computation patterns in the network model.

1) INTER-CHANNEL PARALLEL COMPUTING

Within a convolutional layer, multiple channels share the same input data, and those two- or three-dimensional convolution kernels of each channel are independently applied. Therefore, the multi-channel convolution between input data and kernels could be executed in parallel as shown in Fig. 1.

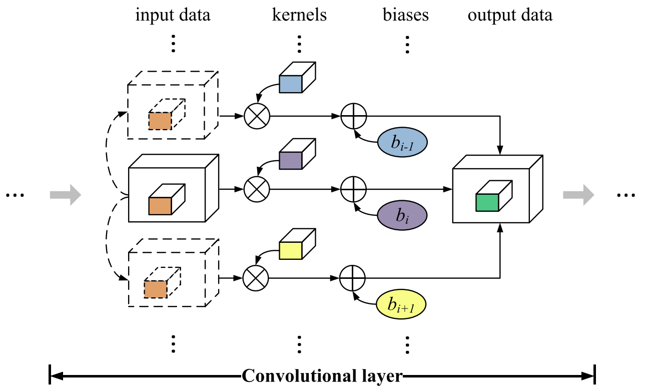

FIGURE 1. Inter-channel parallel computing in the convolutional layer. Dotted-line cuboids in each column denote the shared input data used for the convolution in parallel, and circled times signs denote convolution operations.

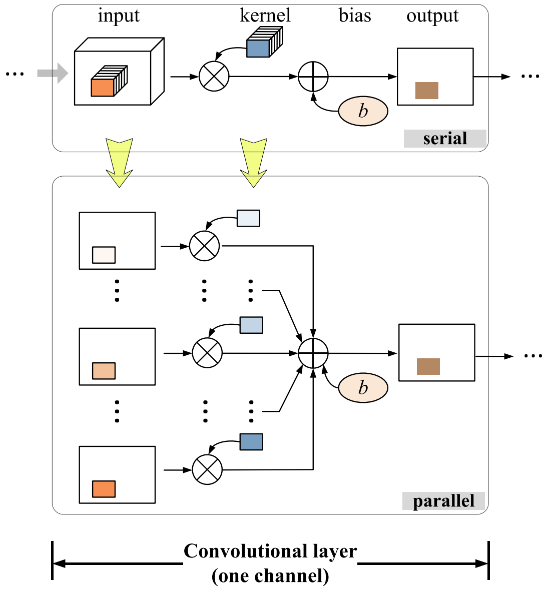

FIGURE 2. In-channel parallel computing in the convolutional layer.

For a given convolutional layer, a series of kernels are convoluted with the identical input data separately to achieve the speedup of computation.

2) IN-CHANNEL PARALLEL COMPUTING

In each channel of a convolutional layer, the depth of kernels is accordant with that of its input data. Based on the process in (9), two parallel schemes, including data parallelism and computation parallelism, are used in the implementation of in-channel convolution. Without loss of generality, let us consider a channel of some middle convolutional layer in the CNN, as shown in Fig. 2. The three-dimensional data are passed in parallel from the previous layer to the current convolutional layer. Then all the two-dimensional data matrices in width and height dimensions are simultaneously convoluted with corresponding kernel slices along the depth direction. Finally, parallel addition of matrices of these results as well as the bias parameter is made to produce the output of the channel.

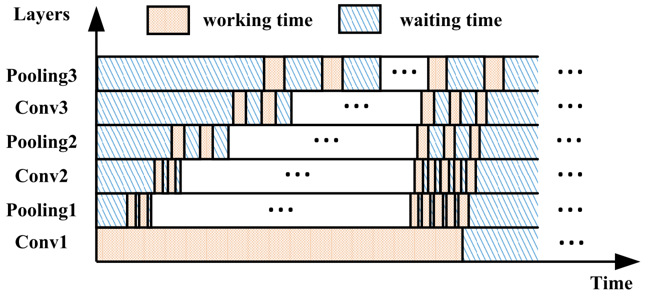

FIGURE 3. Timeline illustration of convolutional and pooling layers execution in the pipeline fashion for one spectrogram input.

3) INTER-LAYER PARALLEL COMPUTING

From the overall viewpoint of CNN model, it could be viewed as a time series structure. The spectrogram of mmW radar mD signals from human targets is transferred and handled sequentially between layers. However, based on the computation patterns in CNN, we could see that there is actually no need to start the execution of some layer until the completion of the previous one. Thus the running of the CNN model is organized in the pipeline fashion. For example, each step of the convolution operation in the convolutional layer only requires a small data patch, which has the same size as the corresponding kernel. The operation in the convolutional layer would be launched when the data patch is produced from the previous one. A similar process is also employed to the implementation of pooling layers. Fig. 3 shows working and waiting time in the execution of convolutional and pooling layers for one spectrogram input. It indicates that such inter-layer parallel computing helps reduce the overall execution time.

4) CONVOLUTION GROUP PARALLEL COMPUTING

The input of pooling manipulation comes from activation results, whose dimensions are identical with the output of previous convolutional layer. According to pooling rules, the cache before the pooling layer may be not needed any more if the data cluster of a pooling region is provided in parallel. On the basis, a convolution group parallel computing is designed to operations in one channel for the optimization of caching usage and time. Fig. 4 illustrate one step process of this parallelism by taking a 2 × 2 pooling as an example. A group of 4 multiplication as well as the addition of bias in one step of the channel in the convolutional layer is calculated at the same time, and results are passed through activation function and pooling layer in parallel.

C. OPTIMIZATION OF CLASSIFICATION DECISION

The softmax would transform output of the last fully connected layer into some numerical representations in the form of probability by using (16). In the classification application,

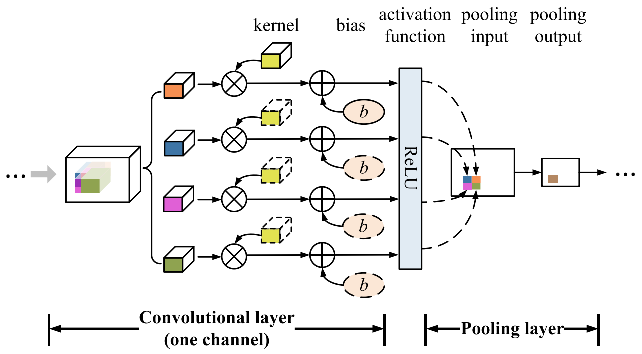

FIGURE 4. Illustration of convolution group parallel computing by taking a 2 × 2 pooling as an example. Dotted-line cuboids denote shared kernels for the multiplication in parallel.

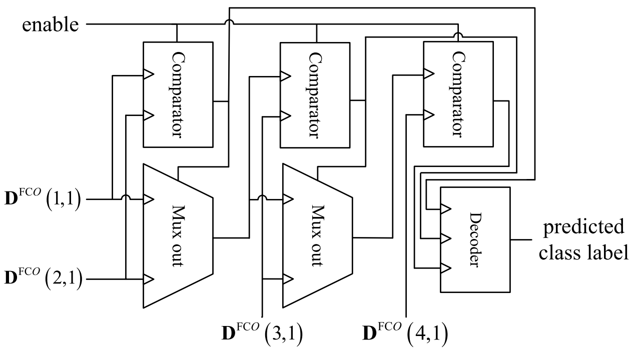

FIGURE 5. Logic diagram of the classification decision in the case of $k = 1, 2, 3$ and 4.

for a given spectrogram of mD signals, the class label of human activity may be predicted corresponding to the maximum value of probabilities η_k. It should be noted in (16) that η_k is monotonically increasing with respect to $\mathbf{D}^{FCO}(k, 1)$. It indicates that the node with maxima of $\mathbf{D}^{FCO}(k, 1)$ also corresponds to the same prediction of class label as using maxima of η_k. Therefore, we could get rid of the softmax function in the classification decision. Then the class label for an input spectrogram would directly be determined by comparing element values of the vector $\mathbf{D}^{FCO}$. Fig. 5 shows the logic diagram of the optimized classification decision when dimensionality of the last fully connected layer output is 4×1. It is obviously beneficial for the computation reduction by removing the softmax step.

V. EXPERIMENTS AND DISCUSSIONS

A. DATASET

To validate the proposed method, some measured mmW radar data from a variety of human activities are collected. We use a Texas Instruments AWR1443 automotive radar sensor [30], which operates at about 77 GHz. The effective bandwidth of radar signals is about 1.536 GHz. Repeat measurements of 9 single persons are carried out. Every individual is positioned 3 meters in front of the mmW radar, and performs 4 actions in place, including walking, jogging, jumping, and walking while holding a stick, respectively. In addition, two aspect angles of human gaiting direction, i.e., 0° and 90°, are considered in every action case. Taking into account the normal periodicity of these human activities, the measurement duration is 2.1 s. The dataset totally consists of 11520 records, as listed in Table 1.

In the STFT processing, a 51.2 ms Hamming window is applied. It slides forward by 20 ms at a time until 100 steps are reached. Hence, according to (8), the size of a mD spectrogram, which acts as the CNN input, is 256×100. Fig. 6 shows some mD spectrogram samples of the person 1 with different

TABLE 1. Numbers of records in the dataset used for experiments.

Actions in place		Walking	Jogging	Jumping	Walking while holding a stick
Persons 1~9	0°	1440	1440	1440	1440
	90°	1440	1440	1440	1440

TABLE 2. Configuration and computational workload of the CNN model.

	Layer	Kernel size	No. of kernels	Output size	No. of operations
L1	Conv / ReLU	5×5×1	4	252×96×4	4935168
L2	Pooling	2×2	--	126×48×4	72576
L3	Conv / ReLU	5×5×4	8	122×44×8	8631744
L4	Pooling	2×2	--	61×22×8	32208
L5	Conv / ReLU	5×5×8	16	57×18×16	6582816
L6	Pooling	2×2	--	28×9×16	12096
L7	FC / Hard Tanh	--	--	100×1	806600
L8	FC / Classification decision	--	--	4×1	803

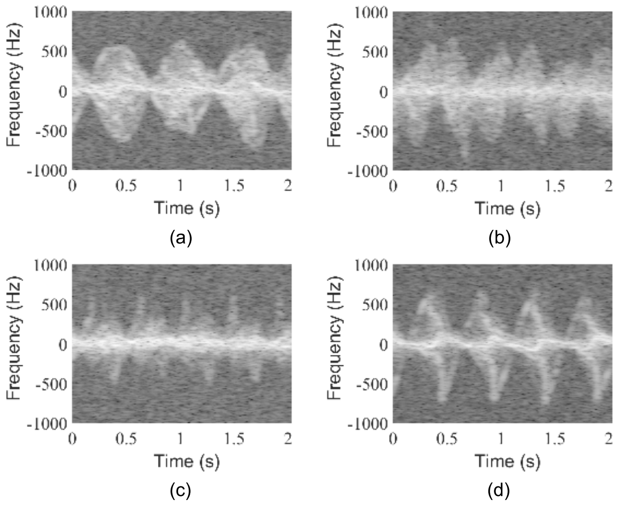

FIGURE 6. MD spectrogram samples of the person 1 with different actions in place at the aspect angle of 0°. (a) Walking. (b) Jogging. (c) Jumping. (d) Walking while holding a stick.

TABLE 3. Classification accuracy of the CNN model on human activities with various training and test data sets.

	Person 1	Person 2	Person 3	Person 4	Person 5	Person 6	Person 7	Person 8	Person 9	Average
Person 1	--	87.66%	82.50%	88.32%	93.40%	93.40%	89.96%	85.23%	86.13%	88.33%
Person 2	87.66%	--	82.23%	91.99%	97.11%	90.78%	84.80%	83.32%	85.31%	87.90%
Person 3	82.50%	82.23%	--	89.77%	88.09%	88.09%	81.80%	75.04%	84.49%	84.00%
Person 4	88.32%	91.99%	89.77%	--	96.76%	94.02%	88.24%	86.91%	88.05%	90.51%
Person 5	93.40%	97.11%	88.09%	96.76%	--	96.80%	92.81%	89.26%	93.32%	93.44%
Person 6	93.40%	90.78%	88.09%	94.02%	96.80%	--	91.02%	85.20%	89.10%	91.05%
Person 7	89.96%	84.80%	81.80%	88.24%	92.81%	91.02%	--	83.28%	82.03%	86.74%
Person 8	85.23%	83.32%	75.04%	86.91%	89.26%	85.20%	83.28%	--	84.57%	84.10%
Person 9	86.13%	85.31%	84.49%	88.05%	93.32%	89.10%	82.03%	84.57%	--	86.63%
Overall average										88.08%

actions in place when he behaves facing toward the radar LOS. We can see that the differences of time-varying mD frequency herein are obvious in the time-frequency domain owing to his normal movements during the measurements. The influence of individual activities on the classification performance is also discussed later in detail.

B. CLASSIFICATION PERFORMANCE ANALYSIS OF THE CNN MODEL

For learning parameters of the CNN model, a training dataset is generated by using some of measured mmW radar data from 7 out of the 9 individuals performing 4 actions in place. The other data from the remaining 2 individuals are used as the test dataset. Table 2 shows the configuration of CNN model used in the work for the human activity classification as well as computational workload for each layer. The operations summarized herein include addition, multiplication and comparison as stated in Section III.

Considering potential impact of training dataset composition, we make a series of experiments with possible groups of training data. Classification results in these cases are listed in Table 3. The value in each cell suggests the classification accuracy of human activities on a test dataset of two target persons in corresponding row and column while spectrograms from the other seven persons are used as the training data. We can see that most of results exceed 80% except the group case of persons 3 and 8, and averaged accuracy over all the cases could achieve about 88.08%. It indicates that the CNN model for mmW radar mD based human activity classification is adaptive to the change of training data sets to some extent.

We would further study the worst case, where data from persons 3 and 8 makes up the test set while the other data are taken as the training set. Tables 4 and 5 elaborate the classification accuracy of four activities for both target persons in the case. With the comprehensive consideration of Table 3, it may be inferred that jogging and jumping in place performed by persons 3 and 8 have much resemblance, but are dissimilar to those of other persons. On the other hand, there is actually little difference of the two actions from both persons 3 and 8 in the data collection procedure, as shown in Fig. 7.

TABLE 4. Confusion matrix of classification results for target person 3 in the worst case.

	Predicted classes			
	Walking	Jogging	Jumping	Walking while holding a stick
Walking	316	0	0	4
Jogging	0	232	88	0
Jumping	44	174	102	0
Walking while holding a stick	95	0	0	225

TABLE 5. Confusion matrix of classification results for target person 8 in the worst case.

	Predicted classes			
	Walking	Jogging	Jumping	Walking while holding a stick
Walking	270	0	49	1
Jogging	21	263	25	11
Jumping	19	89	211	1
Walking while holding a stick	19	0	0	301

C. ACCELERATION PERFORMANCE ANALYSIS OF THE CNN IMPLEMENTATION ON FPGA

According to Section IV.A, a variety of quantization strategies are used for the representation of spectrogram input data, inter-layer input data (i.e., feature maps), as well as weight and bias parameters, as shown in Table 6. Table 7 presents the memory requirement of different types of data in the CNN before and after quantization. We can see that memory resources, most of which are consumed by weight parameters and inter-layer input data, are approximately reduced by a half after quantization. Due to the small number of bias parameters, they are designed to retain the same bitwidth as original ones while the dynamic configuration is still applied for the purpose of data bit alignment.

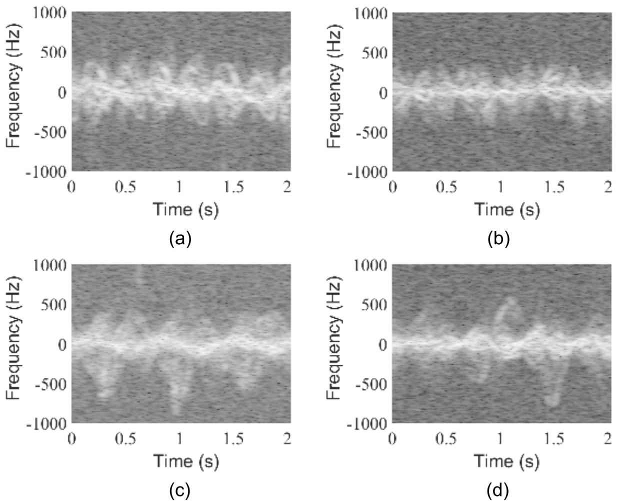

FIGURE 7. MD spectrogram samples of persons 3 and 8 performing jogging and jumping in place, respectively. (a) Jogging in place from the person 3. (b) Jumping in place from the person 3. (c) Jogging in place from the person 8. (d) Jumping in place from the person 8.

TABLE 6. Data quantization used for FPGA based CNN accelerator.

Layers	Data	Bitwidth	Fraction bit	Integer bits	Signed / unsigned
Conv1	Input	16	15	1	Unsigned
	Bias	32	30	1	Signed
Conv2	Input	16	13	3	Unsigned
	Bias	32	28	3	Signed
Conv3	Input	16	9	7	Unsigned
	Bias	32	24	7	Signed
FC1	Input	16	10	6	Unsigned
	Bias	32	25	6	Signed
FC2	Input	16	11	4	Signed
	Bias	32	26	5	Signed
Conv1~3 FC1~2	Weights	16	15	0	Signed

TABLE 7. Comparison of memory requirement before and after quantization.

	Before quantization	After quantization
Spectrogram input	819.20 Kb	409.60 Kb
Inter-layer input	6.24 Mb	3.12 Mb
Bias	4.22 Kb	4.22 Kb
Weights	13.05 Mb	6.52 Mb

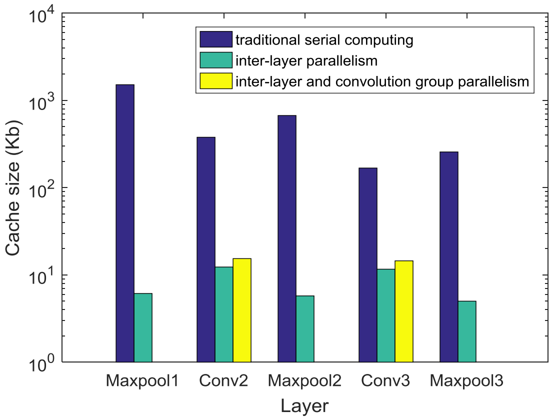

FIGURE 8. Cache usage in convolutional and pooling layers with different parallelism.

TABLE 8. Utilization of logic elements in the FPGA based CNN accelerator.

Logic element	Used	Available	Percentage of usage
Slice registers	48972	437200	11%
Slice LUTs	79637	218600	36%
Block RAM	470	545	86%
DSP48E1s	420	900	46%

TABLE 9. Averaged execution time of proposed CNN accelerator on FPGA for MMW RADAR MD based human activity classification.

Layer	Computation time / μs
Conv1	248.03
Conv2	222.03
Conv3	55.03
The entire network	249.73

To validate the proposed acceleration method for mmW radar mD based human activity classification, we use the Xilinx Zynq XC7Z045 board [31] for the performance assessment. It is implemented using Vivado HLx 2016.4 design tools on the Linux operation system, and the timing constraint is 150MHz. Fig. 8 shows optimized results of cache usage in convolutional and pooling layers by utilizing inter-layer parallelism and convolution group parallelism. There is a slightly higher consumption of on-chip cache in convolutional layers when the convolution group parallelism is introduced, because more cached feature map data are required for multiple convolution operations in parallel, as illustrated in Fig. 4. However, more benefit is obviously obtained from such design, namely that no cache is needed any more in max pooling layers. It means that more efficient usage of cache would be achieved from the overall network perspective by employing both inter-layer and convolution group parallelism. In addition, considering the limitation of DSP resources on the hardware as well as the pipeline fashion between layers, 16 groups of convolution parallel computing are used in the first convolutional layer. Then 4 groups occur in the second and third convolutional layers. It should be emphasized that some waiting time arises in the third convolutional layer in order to buffer enough feature map data for its convolution group parallel computing. Table 8 summarize logic resources on FPGA used for the proposed CNN accelerator.

On the basis, a number of comparative experiments are carried out to further verify the effectiveness of proposed acceleration method. First, Table 9 lists averaged computation time of every convolutional layer and the entire CNN network when accelerating it on the Xilinx FPGA device.

TABLE 10. Comparison of acceleration performance in FPGA based CNN accelerators.

Methods	LUTs	DSPs in use	Board	GOPs	GOPs/DSP	GOPs/*k*LUTs	Bitwidth	Frequency	GOPs/W
Qiu *et al*. [19]	182616	780	Zynq XC7Z045	136.97	0.18	0.75	16 bit	150 MHz	14.22
Suda *et al*. [20]	120000	727	Straix-V GSD8	117.80	0.16	0.98	16 bit	120 MHz	6.16
Venieris *et al*. [21]	218600	900	Zynq XC7Z045	123.12	0.14	0.56	16 bit	125 MHz	--
Proposed accelerator	81229	420	Zynq XC7Z045	87.03	0.21	1.07	16 bit	150 MHz	21.45

TABLE 11. Comparison of the CNN acceleration on FPGA and GPU.

Parameters	FPGA	GPU
Classification accuracy	87.81 %	88.08 %
Power Consumption	3.37 W	55W
Averaged execution time	249.73 μs	358.91 μs

It is obvious that there exists much runtime overlap between layers as a result of pipeline processing and multiple parallelism strategies. Hence, the CNN execution by using the proposed acceleration method is considerably faster than the serial computing for human activity classification based on mmW radar mD signals. Second, we further evaluate its acceleration performance from the perspective of FPGA implementation and in comparison with some related existing work, as shown in Table 10. Due to the difference of application task, network complexity and FPGA devices with varying available resources, assessment metrics, including giga operations per second per DSP (GOPs/DSP), GOPs per kilo lookup tables (GOPs/*k*LUTs) and GOPs per Watt (GOPs/W), are preferable for their comparison. We can see that the proposed acceleration method achieves higher GOPs/DSP, GOPs/*k*LUTs and GOPs/W. It indicates that the proposed accelerator has higher efficiency of FPGA resource usage than others. Finally, Table 11 summarizes the performance of CNN acceleration on the Xilinx Zynq XC7Z045 board and one NVIDIA GeForce GTX 1080 Ti GPU [32], respectively. The contrast results show that the classification accuracy is only reduced by 0.27% owing to the fixed-point number presentation of model parameters and data. However, the proposed FPGA based CNN accelerator is significantly superior in execution speed and power consumption.

VI. CONCLUSION

In this paper, we present an acceleration method of CNN on FPGA for the mmW radar based human activity classification. First, it takes advantage of the high Doppler sensitivity of mmW radar, and thus uses the mD spectrogram represented in the time-frequency domain as the CNN input. Second, according to the FPGA architecture and the computational process of CNN model, a series of acceleration designs are provided and comprehensively implemented. Since convolution layers consume considerable computation resources, the pipeline parallelism of FPGA is deeply exploited for those inter- and in-channel operations. Moreover, some data quantization and classification decision strategies are adopted herein to further optimize the hardware utilization, memory requirement and power consumption. Finally, a measured radar dataset of real human targets is collected with a 77 GHz FMCW radar for the performance evaluation of the proposed method. Results based on some general assessment metrics on FPGA show that it achieves more efficient usage of the hardware resources. Besides, the proposed acceleration method on FPGA outperforms that on a NVIDIA GPU by about 30.42% in the averaged execution time of CNN in test, and could still reaches the classification accuracy of about 87.81% (i.e., only 0.27% decrease).

REFERENCES

[1] D. K. Barton and S. A. Leonov, *Radar Technology Encyclopedia*. Boston, MA, USA: Artech House, 1997.

[2] M. Kishida, K. Ohguchi, and M. Shono, "79 GHz-band high-resolution millimeter-wave radar," *FUJITSU Sci. Tech. J.*, vol. 51, no. 4, pp. 55–59, Oct. 2015.

[3] 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 Trans. Graph.*, vol. 35, no. 4, Jul. 2016, Art. no. 142.

[4] Infineon Technologies AG. *77 GHz Front-End Radar ICs*. Accessed: Feb. 25, 2019. [Online]. Available: https://www.infineon.com/cms/en/product/sensor/radar-sensor-ics/77ghz-front-end-radar-ics/

[5] Texas Instruments Incorporated. *MmWave Sensors*. Accessed: Dec. 3, 2018. [Online]. Available: http://www.ti.com/sensors/mmwave/overview.html

[6] Continental AG. *Radars*. Accessed: Jan. 8, 2019. [Online]. Available: https://www.continental-automotive.com/en-gl/Passenger-Cars/Chassis-Safety/Advanced-Driver-Assistance-Systems/Radars

[7] J. Dickmann, J. Klappstein, M. Hahn, N. Appenrodt, H.-L. Bloecher, K. Werber, and A. Sailer, "'Automotive radar the key technology for autonomous driving: From detection and ranging to environmental understanding,'" in *Proc. IEEE Radar Conf.*, Philadelphia, PA, USA, May 2016, pp. 1–6.

[8] K. Mazouni, A. Zeitler, J. Lanteri, C. Pichot, J.-Y. Dauvignac, C. Migliaccio, N. Yonemoto, A. Kohmura, and S. Futatsumori, "76.5 GHz millimeter-wave radar for foreign objects debris detection on airport runways," *Int. J. Microw. Wireless Technol.*, vol. 4, no. 3, pp. 317–326, Jun. 2012.

[9] M. G. Amin, Y. D. Zhang, F. Ahmad, and K. C. D. Ho, "Radar signal processing for elderly fall detection: The future for in-home monitoring," *IEEE Signal Process. Mag.*, vol. 33, no. 2, pp. 71–80, Mar. 2016.

[10] B. Vandersmissen, N. Knudde, A. Jalalvand, I. Couckuyt, A. Bourdoux, W. D. Neve, and T. Dhaene, "Indoor person identification using a low-power FMCW radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3941–3952, Jul. 2018.

[11] J. S. Patel, F. Fioranelli, M. Ritchie, and H. Griffiths, "Multistatic radar classification of armed vs unarmed personnel using neural networks," *Evolving Syst.*, vol. 9, no. 2, pp. 135–144, Jun. 2018.

[12] V. C. Chen, F. Li, S.-S. Ho, and H. Wechsler, "Micro-Doppler effect in radar: Phenomenon, model, and simulation study," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 1, pp. 2–21, Jan. 2006.

[13] Y. Kim and H. Ling, "Human activity classification based on micro-Doppler signatures using a support vector machine," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 5, pp. 1328–1337, May 2009.

[14] Y. Kim, S. Choudhury, and H.-J. Kong, "Application of micro-Doppler signatures for estimation of total energy expenditure in humans for walking/running activities," *IEEE Access*, vol. 4, pp. 1548–1557, 2016.

[15] J. A. Nanzer and R. L. Rogers, "Bayesian classification of humans and vehicles using micro-Doppler signals from a scanning-beam radar," *IEEE Microw. Wireless Compon. Lett.*, vol. 19, no. 5, pp. 338–340, May 2009.

[16] Y. Kim and T. Moon, "Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 1, pp. 8–12, Jan. 2016.

[17] H. T. Le, S. L. Phung, A. Bouzerdoum, and F. H. C. Tivive, "Human motion classification with micro-Doppler radar and Bayesian-optimized convolutional neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Calgary, AB, Canada, Apr. 2018, pp. 2961–2965.

[18] B. Jokanović and M. Amin, "Fall detection using deep learning in range-Doppler radars," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 1, pp. 180–189, Feb. 2018.

[19] J. Qiu, J. Wang, S. Yao, K. Guo, B. Li, E. Zhou, J. Yu, T. Tang, N. Xu, S. Song, Y. Wang, and H. Yang, "Going deeper with embedded FPGA platform for convolutional neural network," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays (FPGA)*, Monterey, CA, USA, Feb. 2016, pp. 26–35.

[20] N. Suda, V. Chandra, G. Dasika, A. Mohanty, Y. Ma, S. Vrudhula, J.-S. Seo, and Y. Cao, "Throughput-optimized OpenCL-based FPGA accelerator for large-scale convolutional neural networks," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays (FPGA)*, Monterey, CA, USA, Feb. 2016, pp. 16–25.

[21] S. I. Venieris and C.-S. Bouganis, "Latency-driven design for FPGA-based Convolutional Neural Networks," in *Proc. 27th Int. Conf. Field Program. Logic Appl.*, Ghent, Belgium, Sep. 2017, pp. 1–8.

[22] V. C. Chen, "Doppler signatures of radar backscattering from objects with micro-motions," *IET Signal Process.*, vol. 2, no. 3, pp. 291–300, Sep. 2008.

[23] J. O. Hinz and U. Zölzer, "A MIMO FMCW radar approach to HFSWR," *Adv. Radio Sci.*, vol. 9, pp. 159–163, Jul. 2011.

[24] V. C. Chen and H. Ling, *Time-Frequency Transforms for Radar Imaging and Signal Analysis*. Boston, MA, USA: Artech House, 2002.

[25] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn.*, Haifa, Israel, Aug. 2010, pp. 807–814.

[26] R. Collobert, "Large scale machine learning," Ph.D. dissertation, Dept. Comput. Sci., Univ. Paris VI, Pairs, France, 2004.

[27] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," in *Proc. Int. Conf. Learn. Represent.*, Vancouver, BC, Canada, Apr. 2018, pp. 1–13.

[28] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[29] C. Farabet, B. Martini, P. Akselrod, S. Talay, Y. LeCun, and E. Culurciello, "Hardware accelerated convolutional neural networks for synthetic vision systems," in *Proc. IEEE Int. Symp. Circuits Syst.*, Pairs, France, May 2010, pp. 257–260.

[30] Texas Instruments Incorporated. *AWR1443 Single-Chip 76-GHz to 81-GHz Automotive Radar Sensor Evaluation Module*. Accessed: Feb. 26, 2019. [Online]. Available: http://www.ti.com/tool/AWR1443BOOST

[31] Xilinx Incorporated. *ZC706 Evaluation Board for the Zynq-7000 XC7Z045 SoC*. Accessed: Mar. 12, 2019 [Online]. Available: https://www.xilinx.com/support/documentation/boards_and_kits/zc706/ug954-zc706-eval-board-xc7z045-ap-soc.pdf

[32] NVIDIA Corporation. *Geforce GTX 1080 Ti*. Accessed: Nov. 5, 2018. [Online]. Available: https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1080-ti/

PENG LEI received the B.S. and Ph.D. degrees in electrical engineering from Beihang University, Beijing, China, in 2006 and 2012, respectively, where he is currently an Assistant Professor with the School of Electronic and Information Engineering.

His research interests include signal processing, especially in time–frequency analysis and spectral estimation, image processing, and target recognition.

Dr. Lei was the recipient of the 2011 IEEE IGARSS Student Travel Grant.

JIAWEI LIANG received the B.S. degree from Beihang University, Beijing, China, in 2019, where he is currently pursuing the M.S. degree in machine learning and its implementation on FPGA with the School of Electronic and Information Engineering.

ZHENYU GUAN (M'17) received the Ph.D. degree in electronic engineering from Imperial College London, U.K., in 2013. Since then, he has been with Beihang University, Beijing, China, as a Lecturer. His current research interests include cryptography engineering, the IoT security, and blockchain. He is a member of the IEICE.

JUN WANG received the B.S. degree from Northwest Ploytechnic University, Xi'an, China, in 1995, and the M.S. and Ph.D. degrees from Beihang University, Beijing, China, in 1998 and 2001, respectively, where he is currently a Professor with the School of Electronic and Information Engineering.

His research interests include signal processing, DSP/FPGA real-time architecture, and target recognition and tacking.

TONG ZHENG received the M.S. degree from the North China University of Technology (NCUT), Beijing, China, in 2017. She is currently pursuing the Ph.D. degree in machine learning and its application in SAR images with the School of Electronic and Information Engineering, Beihang University.