SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

August 2024, Vol. 67, Iss. 8, 182401:1–182401:21
https://doi.org/10.1007/s11432-023-3958-4

A comprehensive analysis of DAC-SDC FPGA low power object detection challenge

Jingwei ZHANG†, Guoqing LI†, Meng ZHANG*, Xinye CAO, Yu ZHANG, Xiang LI, Ziyang CHEN & Jun YANG

School of Electronics Science and Engineering, Southeast University, Nanjing 210096*, China*

Received 20 September 2023/Revised 5 December 2023/Accepted 20 February 2024/Published online 24 July 2024

Abstract The lower power object detection challenge (LPODC) at the IEEE/ACM Design Automation Conference is a premier contest in low-power object detection and algorithm (software)-hardware co-design for edge artificial intelligence, which has been a success in the past five years. LPODC focused on designing and implementing novel algorithms on the edge platform for object detection in images taken from unmanned aerial vehicles (UAVs), which attracted hundreds of teams from dozens of countries to participate. Our team SEUer has been participating in this competition for three consecutive years from 2020 to 2022 and obtained sixth place respectively in 2020 and 2021. Recently, we achieved the championship in 2022. In this paper, we presented the LPODC for UAV object detection from 2018 to 2022, including the dataset, hardware platform, and evaluation method. In addition, we also introduced and discussed the details of methods proposed by each year's top three teams from 2018 to 2022 in terms of network, accuracy, quantization method, hardware performance, and total score. Additionally, we conducted an in-depth analysis of the selected entries and results, along with summarizing representative methodologies. This analysis serves as a valuable practical resource for researchers and engineers in deploying the UAV application on edge platforms and enhancing its feasibility and reliability. According to the analysis and discussion, it becomes evident that the adoption of a hardware-algorithm co-design approach is paramount in the context of tiny machine learning (TinyML). This approach surpasses the mere optimization of software and hardware as separate entities, proving to be essential for achieving optimal performance and efficiency in TinyML applications.

Keywords tiny machine learning, object detection, convolutional neural networks, algorithm-hardware co-design, low power, field programmable gate array

1 Introduction

In recent years, artificial intelligence (AI) algorithms, represented by deep neural networks (DNNs), have obtained significant success in the computer vision field [1,2], like VGG [3], ResNet [4,5], DenseNet [6], and Dense2Net [7]. The majority of DNNs were implemented on high-performance graphics processing unit (GPU) servers for high inference speed. However, edge AI applications constrained by severely limited computational and memory resources are also important applications [8, 9]. These applications demand not only exceptional DNN inference accuracy but also rapid inference speeds, high throughput, and energy efficiency to address real-world requirements effectively [10–14]. The 2018 System Design Contest (SDC) organized by the IEEE/ACM Design Automation Conference (DAC) promotes the development of edge AI [15]. DAC-SDC is the top contest in software (algorithm) and hardware co-design to accelerate DNNs [16] on the edge platform. From 2018 to 2022, the DAC-SDC has held five sessions successfully. Our SEUer team won the championship in DAC-SDC 2022 and obtained the 6th place in 2020 and the 6th place in 2021 [17, 18]. This paper provides a detailed review and analysis of DAC-SDC.

The lower power object detection challenge (LPODC) centered around the development and deployment of innovative algorithms for object detection within images captured by unmanned aerial vehicles (UAVs) [16]. Real-time processing and low power consumption are very important for UAV platforms [19, 20]. In contrast to conventional computer vision challenges like ImageNet [21] and the PASCAL

* Corresponding author (email: zmeng@seu.edu.cn)
† Zhang J W and Li G Q have the same contribution to this work.

VOC dataset [22], LPODC not only focused on accuracy, but also evaluated the inference speed and power consumption on the same edge low-power platform.

The released dataset in LPODC contains 150k images provided by a DJI UAV company[1)]. The dataset comprises images obtained from real UAVs, thus accurately depicting the practical scenarios and challenges encountered in UAV applications. A single object of interest needs to be detected in the LPODC task, so IoU (intersection over union) is the accuracy metric. LPODC provided two hardware platforms, an embedded GPU (Jetson TX2 from Nvidia) and a field programmable gate array (FPGA) system on chip (SOC) in 2018 and 2019. In particular, the FPGA SOC was the PYNQ Z-1 board of Xilinx in 2018 and the FPGA SOC was the Ultra96 V1 board in 2019. LPODC only provided the Ultra96 FPGA platform after 2019. The contest on the GPU platform was canceled from 2020 and the FPGA platform was changed to the Ultra96 V2 board from 2020.

The DAC-SDC is open to both industry and academia. Each year dozens of teams from all over the world participate in the contest. The participating teams conducted model training using the training dataset provided and subsequently deployed these trained models on the hardware platform furnished by DAC-SDC. The results including frames per second (FPS), energy consumption (Joule, J), and IoU were tested by the DAC-SDC organizer. The evaluation methods were different from 2018 to 2022, which is detailed in Subsection 2.3. The organizer furnished teams with an official evaluation at the conclusion of every month, complete with specific rankings. The ultimate ranking was unveiled at the competition's conclusion, with the Top-3 teams earning invitations to showcase their work during a dedicated technical session at DAC.

Although the organizing committee annually releases the code of the winning entries, there have been only two typical research attempts to analyze it scientifically over the past six years. Xu et al. [15] conducted a relatively comprehensive analysis covering the dataset and object detection tasks of the competition. However, their analysis was limited to the representative achievements of the first edition in 2018, without follow-up exploration of subsequent developments. As participating teams delve deeper into their research, recent achievements have greater potential for overall improvements in object detection task performance. Therefore, conducting in-depth studies on these recent accomplishments adds greater research value. On the other hand, Jia et al. [16], despite extending their study period up to the latest in 2022, faced constraints due to the two-page limit, resulting in a brief overview. In comparison, our work involves a comprehensive analysis of all winning solutions over the past five years, providing detailed summaries of various network models and categorizing hardware structures into two major classes, and discussing their advantages and disadvantages. Furthermore, we delve into details from multiple perspectives, including quantization methods, accuracy, and overall scores, summarizing the advantages of representative methods. Ultimately, considering different task scenarios, such as high-precision or low-power, we propose algorithm-hardware co-design strategies tailored for each scenario.

In this paper, the details of LPODC are first described in Section 2 including the dataset, the hardware platforms, and the evaluation method. The contestants will compete in two different categories: FPGA and GPU. Contests on the GPU platform were only held in 2018 and 2019, and the GPU contest has been canceled since 2020. Xu et al. [15] have reviewed the GPU contest of 2018 in detail. Therefore, the Top-3 designs on the FPGA platform from 2018 to 2022 are introduced in this paper, and the GPU contest is excluded. Subsequently, a comprehensive discussion of the network architecture, quantization, and hardware accelerator of these designs is included, which offers valuable insights and contributes to a deeper comprehension of this domain, thereby advancing the development of object detection algorithms, particularly in UAV applications.

2 LPODC dataset, hardware platform and evaluation criteria

2.1 Dataset

The LPODC task is the single object detection task, one of the most crucial tasks in UAV applications [19, 20]. The dataset is from DJI company, which contains 12 categories (see Figure 1) and 95 subcategories. The complete DJI-UAV dataset is partitioned into three segments: the training dataset, comprising 100000 images containing objects of interest; the validation dataset, consisting of 1000 images; and the concealed test set, comprising 52500 images exclusively accessible to the contest organizers for official

1) Dji-uav dataset. https://byu.box.com/s/hdgztcu12j7fij397jmd68h4og6ln1jw.

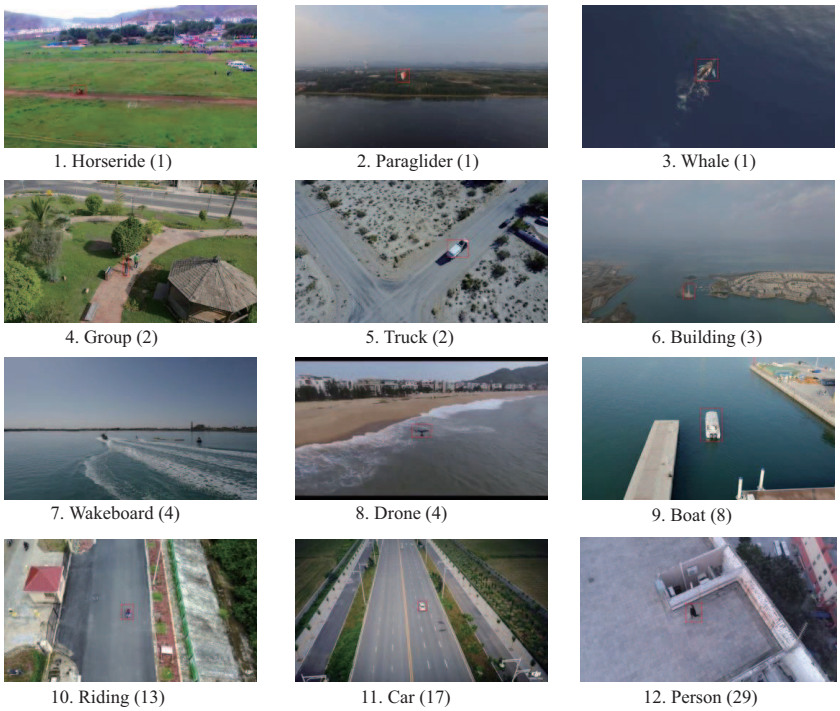

Figure 1 (Color online) Overview of the DJI-UAV dataset. There are 12 categories. Note that there is only one object in each image and the object size ratio is very small.

evaluation. Xu et al. [15] conducted an analysis of the distributions pertaining to category, object size ratio, and image brightness within both the training and testing datasets. The categories person, car, and rider exhibit a higher image count in comparison to other categories, primarily due to the inclusion of a greater number of subcategories within them. Additionally, a significant proportion of images within the dataset feature object sizes ranging from 1% to 2% of the dimensions of the captured images (640x360). Nonetheless, it is worth noting that the average object size ratio in ILSVRC [21] stands at 17%, and in PASCAL VOC [22], it registers at 20%. The primary feature of UAV view images is the presence of a small object size ratio, which presents a greater detection challenge compared to medium and large objects. The majority of the images exhibit a moderate level of brightness and contain a moderate amount of information. Conversely, there are notably fewer images characterized by extreme levels of brightness and information content, resulting in a distribution reminiscent of a Gaussian distribution.

2.2 Hardware platforms

In the context of LPODC, FPGA and GPU types of platforms were made available to the participating teams, with Xilinx and Nvidia being responsible for each platform, respectively. The GPU platform is Nvidia Jetson TX2 as shown in Figure 2(a), which is an embedded AI accelerating device. It is suitable for UAV applications, which can provide more than 1T FLOPS (floating-point operations per second) of FP16 computing performance while consuming less than 7.5 watts of power. After 2019, the contests on GPU platforms were canceled.

Three types of FPGA platform are adopted from 2018 to 2022. In 2018, the FPGA platform is the Xilinx PYNQ Z1 board as shown in Figure 2(b), which contains both programmable logic (PL, ZYNQ XC7Z020) and processing system (PS) with low-power central processing unit core (Cortex-A9 processor). As presented in Table 1, the XC7Z020 FPGA chip contains 53k 6-input lookup tables (LUT), 220 DSPs, 4.9 MB fast block random access memory (BRAM), and 106k flip-flop (FF). A 512 MB third-generation synchronous dynamic random access memory (DDR3 SDRAM) with 16-bit bus at 1050 Mbps can be accessed by both PS and PL. In 2019, as shown in Figure 2(c) a more advanced Ultra96 V1 board is adopted, which has a more advanced Cortex-A53 processor. It also adopted the more advanced ZU3EG FPGA chip with more advanced 16 nm technology. The processor and FPGA can access 2 GB of DDR4 memory. The ZU3EG chip has more hardware resources than the XC7Z020 FPGA chip. As presented in Table 1, the ZU3EG FPGA chip contains 71k 6-input LUT, 360 DSPs, 7.6 MB fast BRAM and 141k FF.

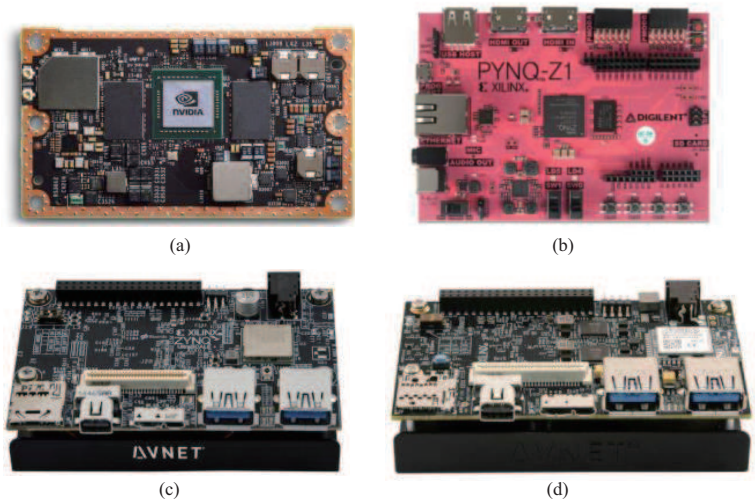

Figure 2 (Color online) Total score of the champion team for each year from 2018 to 2022 according to the evaluation method of 2022. (a) Nvidia Jetson TX2; (b) PYNQ Z1; (c) Ultra96 V1; (d) Ultra96 V2.

Table 1 Comparison of PYNQ Z1 board and Ultra96 board

Board	PYNQ Z1	Ultra96 V1	Ultra96 V2
Processor	Cortex-A9	Cortex-A53	Cortex-A53
Technology (nm)	28	16	16
DRAM	DDR3 512 M	DDR4 2 G	DDR4 2 G
FPGA	XC7Z020	ZU3EG	ZU3EG
LUT	53k	71k	71k
DSP	220	360	360
BRAM (MB)	4.9	7.6	7.6
FF	106k	141k	141k

From 2020 to 2022, the Ultra96 V2 board was adopted. Ultra96-V2 updates and updates the Ultra96-V1 product that was released in 2018. They used the same ZU3EG FPGA chip ZU3EG. The WiFi module and power regulators on the board have been altered, and this is expected to lead to divergent power consumption measurements compared to an identical design operating on the V2 board. In 2019, a 2 A power supply was provided. However, if a design utilizes multiple software cores along with a significant portion of FPGA fabric, this might prove inadequate. Consequently, starting in 2020, a more robust 4 A power supply has been made available to accommodate these requirements.

2.3 Evaluation method

The assessment criteria for the challenge encompass detection accuracy, throughput, and energy consumption. Specifically, object detection accuracy is quantified using the IoU metric. The challenge exclusively focused on IoU results and did not take into account object categories. Consider two bounding boxes (BB): a predicted BB_p and a ground truth BB_g. In that case, the precision or IoU of the predicted bounding box can be defined as the ratio between the area of the union of the predicted bounding box and the ground truth bounding box, divided by the area of the overlap shared by both the predicted bounding box and the ground truth bounding box, as follows:

$$\mathrm{IoU} = \frac{\mathrm{BB}_p \cap \mathrm{BB}_g}{\mathrm{BB}_p \cup \mathrm{BB}_g}. \tag{1}$$

The throughput metric is FPS. Since the number of evaluation images is fixed, the FPS can be calculated as (2), where N is the number of evaluation images and T is the whole runtime for all images. All participating teams use the same official code to complete the run-time calculation.

$$\mathrm{FPS} = N/T. \tag{2}$$

Energy consumption is the energy consumed to compute all the images tested. A power meter is used on the FPGA board during the evaluation. Prior to 2020, power consumption was sampled at a frequency of 100 Hz. Starting from 2020, the sampling frequency for power consumption was adjusted to 20 Hz. Ultimately, the energy consumption is calculated by multiplying the power consumption by the runtime.

There were two different versions to calculate the total score (TS_i), with 2021 as the dividing line. Before 2021, the design was evaluated mainly by accuracy and energy consumption if throughput was larger than the threshold. In other words, throughput was a penalty part of the evaluation. From 2021, accuracy and throughput together constituted the penalty part. Therefore, energy consumption solely decided the final score once the accuracy and throughput were larger than the requirement value.

The total score before 2021 was calculated using (3), which was the product of three parts. For the team i, $\overline{\mathrm{IoU}_i}$ refers to the average IoU result of all images evaluated. FPS_i was the FPS of the team i and $\mathrm{FPS}_{\mathrm{th}}$ was the minimum speed requirement, once FPS_i was lower than $\mathrm{FPS}_{\mathrm{th}}$, this part would be a penalty part. In 2018, $\mathrm{FPS}_{\mathrm{th}} = 5$. In 2019 and 2020, $\mathrm{FPS}_{\mathrm{th}} = 10$. From 2021, $\mathrm{FPS}_{\mathrm{th}} = 30$. ES_i was the energy consumption score of the team i, ES_i could be calculated by the energy consumption of the team i (E_i) and the average energy consumption of the team of participants ($\overline{E}$) as shown in (3).

$$\begin{aligned} \mathrm{TS}_i &= \overline{\mathrm{IoU}_i} \times (\min(\mathrm{FPS}_i, \mathrm{FPS}_{\mathrm{th}})/\mathrm{FPS}_{\mathrm{th}}) \times (1 + \mathrm{ES}_i), \\ \mathrm{ES}_i &= \max\left\{0, 1 + 0.2 \times \log_2 \frac{\overline{E}}{E_i}\right\}. \end{aligned} \tag{3}$$

The total score from 2021 was calculated by (4) which was still the product of three parts. However, IoU and FPS were both part of the penalty in the formula, where 0.7 and 30 are the thresholds for IoU and FPS. The energy consumption score was represented straightly by the energy measured after the logarithmic operation and did not depend on the average power consumption of all teams. Furthermore, the calculation of runtime was slightly different in 2021 and 2022. In 2022, to free the limitation of image loading, runtime only calculated the PL working time rather than the whole system working time.

$$\mathrm{TS}_i = \frac{10^2}{\log_2 E_i} \times \max\{\mathrm{ReLU}(1 - 5 \times \mathrm{ReLU}(0.7 - \overline{\mathrm{IoU}_i})), 0.1\} \times \mathrm{ReLU}\left(1 - \mathrm{ReLU}\left(1 - \frac{\mathrm{FPS}_i}{30}\right)\right). \tag{4}$$

3 TOP-3 designs on FPGA platform from 2018 to 2022

In this section, the Top-3 designs from 2018 to 2022 are introduced according to the timeline.

3.1 2018's

Out of the 61 participating teams, a total of seven teams effectively deployed their designs on the provided FPGA platform. The Top-3 teams are TGIIF, SystemsETHZ, and iSmart2.

TGIIF [23] team from Tsinghua University in Beijing, China proposed an optimized VGG+SSD network. They reduced the depth of VGG [3] and SSD [24] networks by removing some convolutional layers. The optimized network structure proposed by TGIIF is shown in Figure 3(a). The proposed network had 2365k weights and 4006M multiplications. It adopted 8-bit fixed-point activation and 8-bit fixed-point weight. The weight size was reduced to 18.04 MB from 72.17 MB by quantization. TGIIF utilized the Verilog hardware description language for efficient FPGA designs. The FPGA accelerator adopted the single computation engine architecture (systolic array).

As shown in Figures 4(c) and (d), in the structure of the single computing engine (CE), the entire accelerator comprises a single hybrid CE. Unlike dedicated CEs in the streaming structure, this type of CE needs to support multiple scenarios. Therefore, it must simultaneously support various operator types, such as convolution, depthwise separable convolution (DSC), fully connected layers, pooling, activation, etc. The accelerator configures the CE's functionality through custom instructions and calls for execution to complete the computation of the entire network model. As shown in Table 2, the accelerator used 44.5k LUT, 220 DSP, 3.9 MB BRAM, and 57.5k FF resources, and hardware resource utilisation is indicated in parentheses. It obtained the highest 0.624 IoU, achieved 12.0 FPS, and consumed 18444.2 J energy.

SystemsETHZ [25] team from ETH Zurich in Switzerland proposed a HalfSqueezeNet + YoLO network. HalfSqueezeNet was a variation of SqueezeNet [26] by halving the fire layer in SqueezeNet for optimization. The YOLO [27] architecture was used for the detection. The network structure proposed

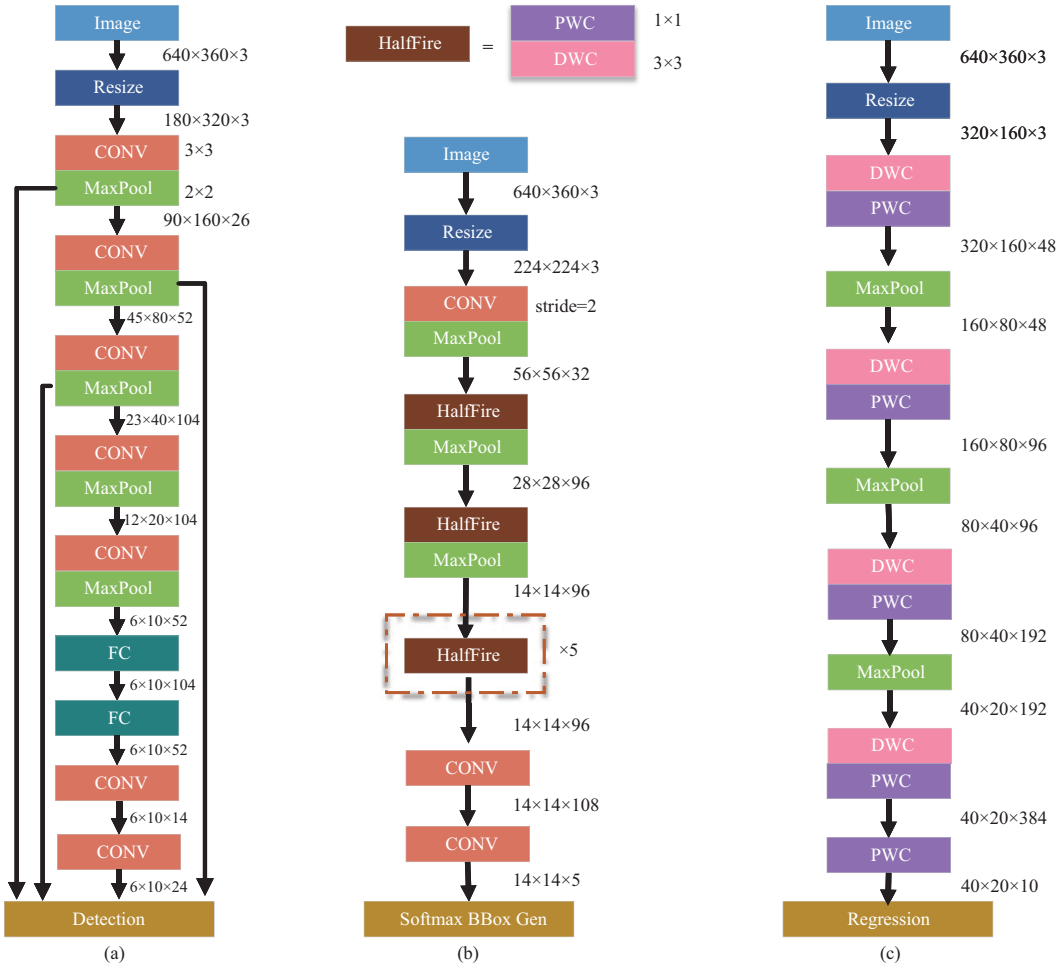

Figure 3 (Color online) Neural network topology of the Top-3 FPGA entries in 2018. (a) TGIIF (SSD); (b) SystemETHZ (HalfSqueezeNet); (c) iSmart2 (MobileNet+YOLO).

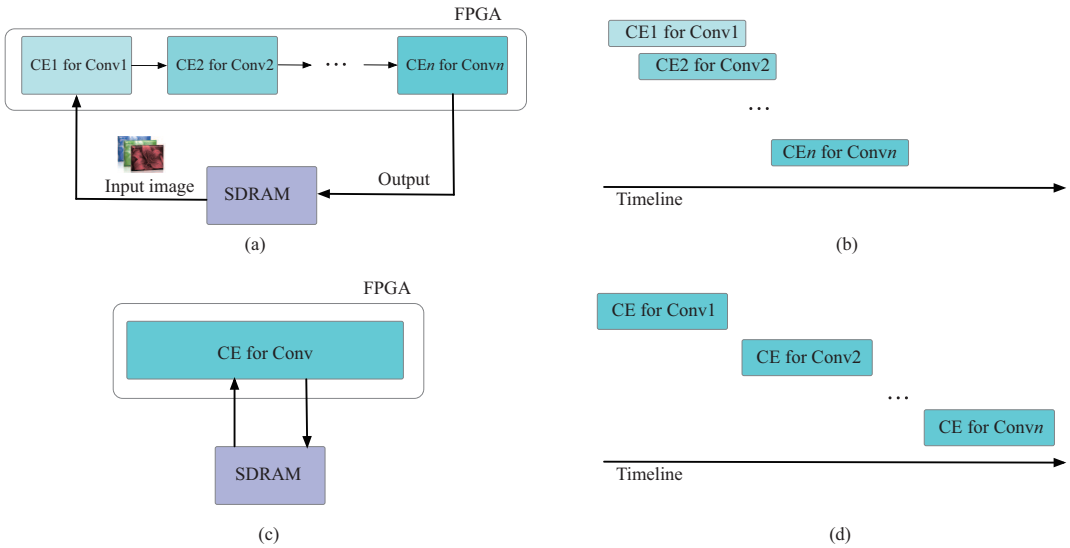

Figure 4 (Color online) Illustration of the streaming architecture accelerator and single CE architecture accelerator. (a) Streaming architecture accelerator; (b) timeline of streaming architecture accelerator; (c) single CE architecture accelerator; (d) time line of single CE architecture accelerator.

by SystemsETHZ is illustrated in Figure 3(b). The network had 18 convolutional layers, 215k weights and

Table 2 Comparison of Top-3 FPGA designs in 2018

Entries	TGIIF	SystemsETHZ	iSmart2
Network	SSD	HalfSqueezeNet	MobileNet+YOLO
Quantization	A8/W8	A5/W1	A16/W8
IoU	0.624	0.492	0.573
FPS	12.0	26.0	7.3
GOPS	56.5	11.2	2.9
Energy (J)	18444.2	4953.2	18502.5
Power (mW)	4200	2450	2590
LUT	44.5k (84.0%)	46.6k (87.9%)	33.4k (63.0%)
DSP	220 (100%)	172 (78.1%)	190 (86.4%)
BRAM (MB)	3.9 (79.6%)	3.8 (77.6%)	4.6 (93.9%)
FF	57.5k (54.2%)	65.7k (62.0%)	23.3k (22.0%)

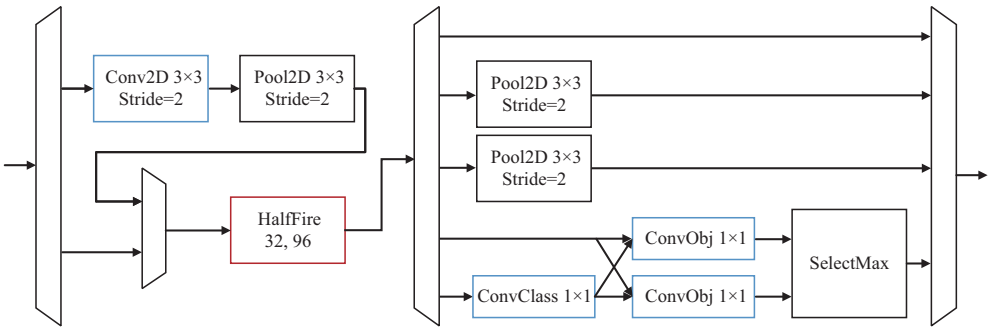

Figure 5 (Color online) Illustration of the internal structure of the CE module of the streaming architecture.

174M multiplication. The approach employed dynamic precision weights across various layers, utilizing a five-bit fixed-point activation format in all activation layers. However, in the first layer, where the input RGB image data is represented as 8-bit, an eight-bit fixed-point format was utilized. The weights in all fire layers were binary [28]. The weight size was reduced to 0.21 MB from 6.56 MB by binary quantization. The entry employed high-level synthesis to facilitate rapid FPGA implementation.

The accelerator adopted a single CE architecture between different Fire modules, but it was a streaming architecture in a Fire module. This particular structure was not a complete streaming structure; it could be referred to as an intra-layer streaming. The advantage was in the fact that, compared to the streaming structure, it could support a larger network with a smaller on-chip storage capacity. Figure 5 illustrates the internal structure of the CE module of the streaming architecture. Pipelined computation does not require completion in a single step. It involves guiding the data flow by switching demux and mux, completing a portion of the computation. The process is repeated through successive switching operations until the entire network computation is complete. This collapsed computation approach enables the data to traverse distinct segments of the neural network as they progress within the architecture. Consequently, this design strategy leverages layer-to-layer parallelism via pipelining, facilitating their simultaneous execution. As shown in Table 2, the accelerator used 46.6k LUT, 172 DSP, 3.8 MB BRAM, and 65.7k FF resources. It obtained 0.492 IoU, achieved the highest 26.0 FPS, and consumed the least 4953.2 J energy.

iSmart2 [29] team from the University of Illinois at Urbana-Champaign (UIUC) in the United States adopted lightweight DSC to construct a variation on MobileNet + YoLO networks [27, 30]. The network structure proposed by iSmart is illustrated in Figure 3(c). The proposed network had 104k weights and 198M multiplication. It adopted quantization to reduce model size: 16-bit fixed point activation and 8-bit fixed point weights. The weight size was reduced to 0.79 MB from 3.17 MB by quantization. The entry employed high-level synthesis to facilitate rapid FPGA implementation. There were two CEs in the accelerator, but no streaming architecture. DSC consists of depthwise convolution (DWC) and pointwise convolution (PWC). One CE was for computing DWC and the other for computing PWC. Two CEs cannot run in parallel. Therefore, it also belongs to a single CE architecture. As shown in Table 2, the accelerator used 33.4k LUT, 190 DSP, 4.6 MB BRAM, and 23.3k FF resources. It obtained 0.573 IoU, achieved 7.3 FPS, and consumed 18502.5 J energy.

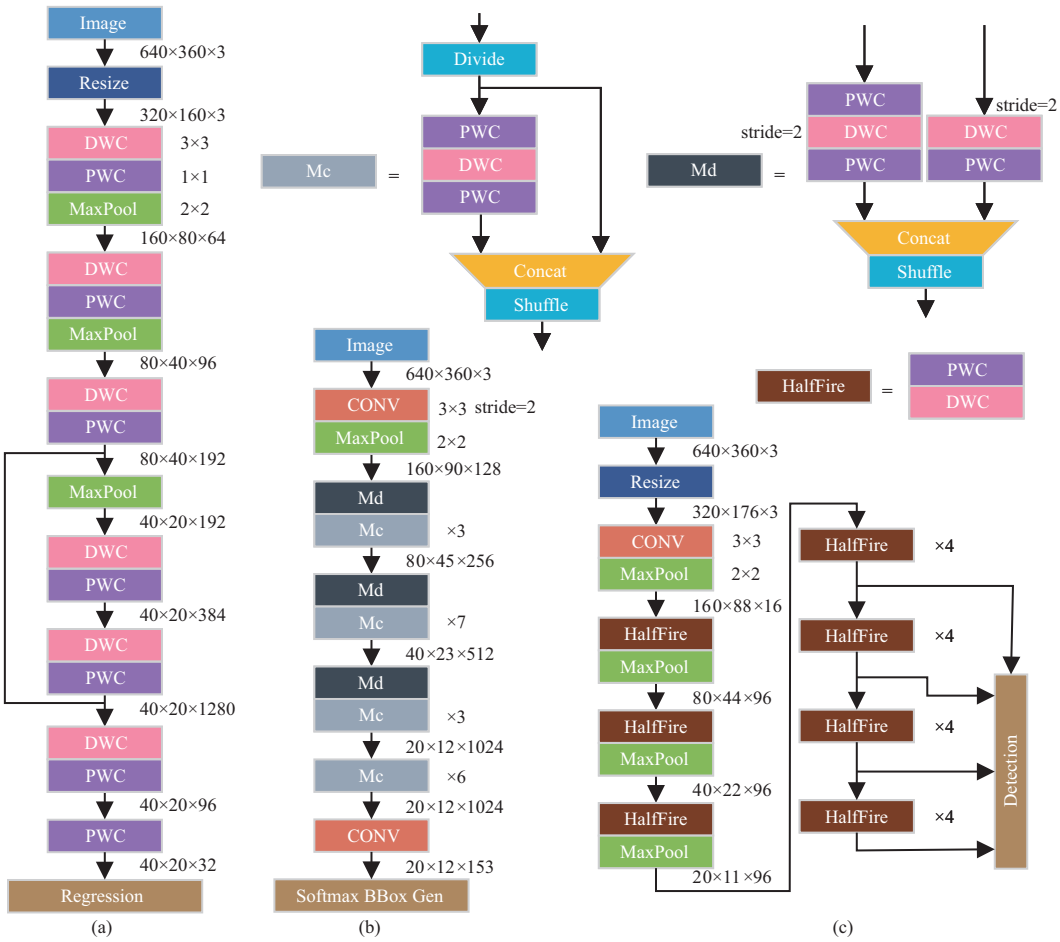

Figure 6 (Color online) Neural network topology of the Top-3 FPGA entries in 2019. (a) iSmart3 (SkyNet); (b) XJTU-Tripler (ShuffleDet); (c) SystemETHZ (RectHalfSqzNet).

3.2 2019's

Out of the 58 participating teams, a total of 11 teams effectively deployed their designs on the provided FPGA platform. The Top-3 teams are iSmart3, XJTU-Tripler, and SystemsETHZ.

iSmart3 [31–34] team proposed an efficient SkyNet using the bottom-up design approach, which had better performance than the network they proposed in 2018. Key members of the iSmart3 team were aligned with key members of iSmart2. As shown in Figure 6(a), SkyNet used a bypass approach to extract the multi-level feature for high IoU and adopted DSC to reduce the model size. The proposed network had 440k weights and 463M multiplication. It adopted quantization to reduce the model size: 9-bit fixed activation layers and 11-bit fixed point weights in all layers. The weight size is reduced to 4.61 MB from 13.43 MB by quantization.

The entry employed high-level synthesis to facilitate rapid FPGA implementation. Like iSmart2, iSmart3 also belongs to a single CE architecture, and the general architecture of iSmart3 of the low-power object detection system is illustrated in Figure 7. The architecture is based on a heterogeneous architecture that contains PL and a high-performance PS. PL comprises four modules. A memory arbitration module (ARBI) is employed for PL to directly access external memory via the AXI bus. It strategically schedules double data buffers to establish a ping-pong structure, thus optimizing the utilization of on-chip memory resources. The computational module (COMP) encompasses multiple processing engines that are stacked to efficiently handle both PWC and DWC computations. Furthermore, a pooling and activation (PA) module is responsible for managing the PA computations. To ensure seamless coordination, a dedicated control module (CTRL) is meticulously designed to govern the logic and timing of PL, ensuring the appropriate reuse of each module at the precise moment. PS leverages multi-process optimization techniques to enhance the computational speed of image pre-processing and

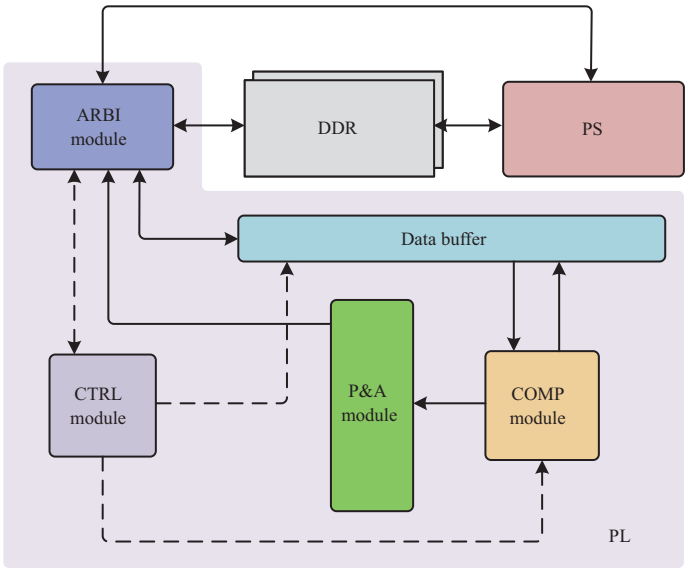

Figure 7 (Color online) Illustration of heterogeneous architecture including PL and PS.

Table 3 Comparison of Top-3 FPGA designs in 2019

Entries	iSmart3	XJTU-Tripler	SystemETHZ
Network	SkyNet	ShuffleDet	RectHalfSqznet
Quantization	A9/W11	A8/W8	A5/W1
IoU	0.716	0.615	0.553
FPS	25.1	50.9	55.1
GOPS	23.2	407.0	73.4
Energy (J)	15215.6	9536.8	6366.1
Power (mW)	7260	9248	6685
LUT	54.5k (76.8%)	45.9K (64.6%)	54.3k (76.5%)
DSP	329 (91.4%)	353 (98.1%)	263 (73.1%)
BRAM (MB)	7.4 (97.4%)	6.9 (90.8%)	6.5 (85.5%)
FF	59.9k (42.5%)	57.9k (41.2%)	74.8k (53.0%)

post-processing procedures. As shown in Table 3, the accelerator used 54.5k LUT, 329 DSP, 7.4 MB BRAM, and 59.9k FF resources. It obtained the highest 0.716 IoU, achieved 25.1 FPS, and consumed 15215.6 J energy.

XJTU-Tripler [35, 36] team from Xi'an Jiaotong University in Xi'an, China proposed a ShuffleDet (variation of ShuffleNetV2 + YoLO.) network. The ShuffleNetV2 was first pre-trained on the ImageNet dataset, and then the feature extracting part was transferred to ShuffleDet. The network structure proposed by XJTU-Tripler is illustrated in Figure 6(b). The network has 8257k weights and 3997M multiplications, which is a little larger than ShuffleNetV2 1×. It also adopted quantization to reduce the size of the model. Both activation and weight were quantized to an 8-bit fixed point. The weight size was reduced to 63.00 MB from 251.98 MB by quantization.

XJTU-Tripler chose to utilize the Verilog hardware description language to achieve an efficient FPGA design. Like SystemsETHZ, the accelerator adopted the single computation engine architecture between different blocks, but the streaming architecture in a block. In particular, their architecture is called HiPU which is mainly used for application-specific integrated circuits (ASIC), with appropriate cuts made to accommodate the resources of ZU3. HiPU supports a variety of common NN operations, operations in the channel direction, matrix operations, vector operations, and scalar operations. Additionally, HiPU is maintained to operate at 233 MHz with a system peak power of 268 GOPS. As shown in Table 3, the accelerator used 45.9k LUT, 353 DSP, 6.9 MB BRAM, and 57.9k FF resources. It obtained 0.615 IoU, achieved 50.9 FPS and consumed 9536.8 J of energy.

SystemsETHZ [25] team came second in 2018 and third in 2019. They proposed RectHalfSqznet based on HalfSqueezent, which is a variation of SqueezeNet+SSD. In order to improve the IoU, SSD

detection architecture was used for multi-level features. The network topology of RectHalfSqznet is shown in Figure 6(c). The proposed network had 588k weights and 66M multiplications. It adopted quantization to reduce model size: 5-bit fixed-point activation and 1-bit fixed-point weights in all layers. The weight size was reduced to 0.56 MB from 17.04 MB by quantization. The entry employed high-level synthesis to facilitate rapid FPGA implementation. As shown in Figure 5, the accelerator adopted a single CE architecture between different Fire modules in 2019, consistent with the design in 2018. Within each Fire module, a streaming architecture was adopted. As shown in Table 3, the accelerator used 54.3k LUT, 263 DSP, 6.5 MB BRAM, and 74.8k FF resources. It obtained 0.553 IoU, achieved the highest 55.1 FPS, and consumed the least 6366.1 J energy due to efficient binary quantization.

3.3 2020's

Out of the 80 participating teams, a total of 13 teams effectively deployed their designs on the provided FPGA platform. The Top-3 teams are BJUT_Runner, Skrskr, and iSmart. Our SEUer team obtained the 6th place in 2020.

BJUT_Runner [37–39] team from Beijing University of Technology in Beijing, China proposed an UltraNet by optimizing the VGG + YOLO network. They reduced the depth of the VGG network by removing some convolutional layers. The optimized network structure proposed by BJUT_Runner is illustrated in Figure 8(a). UltraNet consisted of only 8 standard convolutional layers with kernel size 3×3 and one PWC layer. The proposed network had 210k weights and 200M multiplications. Comparing with previous designs, they proposed a more refined quantization method: 8-bit integer parameters in the first convolutional layer because the input RGB image data were 8-bit, 4-bit integer weights in all convolutional layers. The weight size was reduced to 0.80 MB from 6.4 MB by quantization.

BJUT_Runner employed high-level synthesis to facilitate rapid FPGA implementation. From the design, we see a complete streaming structure for the first time, as described in Figure 4(a). All CEs are connected to create a pipeline, as shown in Figure 4(b). Figure 9 shows the details of the streaming accelerator architecture, which adopts a streaming structure that stores all network parameters on the chip to avoid external memory access during the inference phase. The accelerator consists of dedicated processing engines for each layer and input flows seamlessly through the dataflow architecture, enabling parallel computation across all layers. Intermediate features stream in the accelerator, eliminating the need for memory access to feature maps during inference. As shown in Table 4, the accelerator used 55.0k LUT, 360 DSP, 5.3 MB BRAM, and 64.1k FF resources. It obtained 0.656 IoU, achieved the highest 212.7 FPS, and consumed the least 1641.2 J energy due to efficient streaming architecture.

Skrskr [40] team from ShanghaiTech University in Shanghai, China adopted SkyNet which was proposed by the iSmart3 team in 2019. Comparing to the iSmart3 team, they proposed a more refined quantization method: 8-bit unsigned integer activation and 6-bit signed integer weights in all convolutional layers. The weight size was reduced to 2.52 MB from 13.43 MB by quantization. Skrskr employed high-level synthesis to facilitate rapid FPGA implementation. Like iSmart, the accelerator adopted a single computation engine architecture as described in Figure 7. As shown in Table 4, the accelerator used 57.6k LUT, 360 DSP, 6.9 MB BRAM, and 72.2k FF resources. It obtained the highest 0.731 IoU due to the good network and efficient quantization, achieved 52.4 FPS, and consumed 6764.2 J energy.

iSmart[2)] team also used their proposed SkyNet in 2019. This year, they did not change the network and quantization method. The activation was quantized to a signed fixed point of 9 bits, but the activation parameters were always positive numbers, so the sign bit was always zero after the ReLU activation function. When transferring the activation parameters between SDRAM and FPGA chip, only the 8-bit was valid (the sign bit was always zero). Therefore it was sufficient for transferring the valid 8-bit. If transferring 9-bit activation parameters, 16-bit (7-bit are invalid) data width will be used. Only 8-bit data width was enough after ignoring the sign bit of the activation parameters. They reduced half the transfer of activation parameters by this trick. Moreover, they improved the parallelism of the DWC engine. The accelerator was similar to the accelerator proposed last year, as mentioned in Figure 7. As shown in Table 4, the accelerator used 58.21k LUT, 358 DSP, 7.4 MB BRAM, and 64.6k FF resources. It obtained 0.724 IoU, achieved 50.7 FPS, and consumed 7621.9 J energy.

2020 was a year of significant changes for LPODC. The streaming structure emerged, marking a new era, while the single CE structure experienced its final glory. Here, a summary of the advantages and disadvantages of the single CE is provided. In terms of advantages, the deployment threshold is relatively

2) Dac2020-ismart. https://github.com/jgoeders/dac_sdc_2020_designs/tree/master/iSmart.

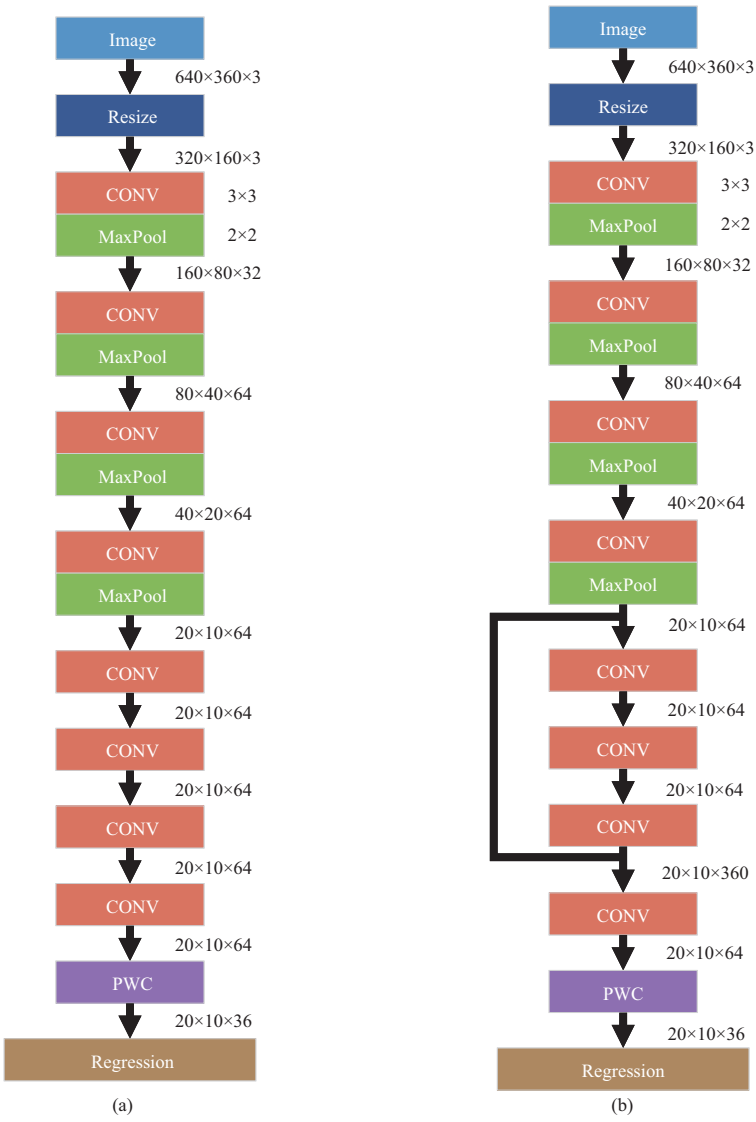

Figure 8 (Color online) Neural network topology of (a) UltraNet and (b) UltraNet_bypass.

low. The main advantage lies in a significantly reduced dependence on memory, allowing the use of external memory in case on-chip memory resources are insufficient. With unrestricted storage capacity, quantization schemes can be chosen based on precision requirements with more flexibility. Large-scale networks are typically better suited for the single CE structure. In disadvantages, there is wastage due to frequent access to memory and inefficient use of computation resources. Since different layers of neural network models have data dependencies, the single CE structure needs to compute and store intermediate feature maps layer by layer. When using on-chip memory, read and write operations for each layer's feature map incur additional memory access latency and power consumption. When using external memory, the transfer time cost of accessing external memory via the bus is several times higher than direct access to on-chip memory. Additionally, during computation in the hybrid CE, usually only one type of operator circuit is in operation, while other functional circuits within the CE are idle, resulting in low computational efficiency for the single CE structure.

3.4 2021's

Out of the 113 participating teams, a total of 25 teams effectively deployed their designs on the provided FPGA platform. The Top-3 teams are Skrskr, iSmart, and SJTU_microe. Our SEUer team also obtained the 6th place in 2020.

Skrskr [41] continued to adopt the SkyNet proposed by the iSmart3 team in 2019. Unlike their design

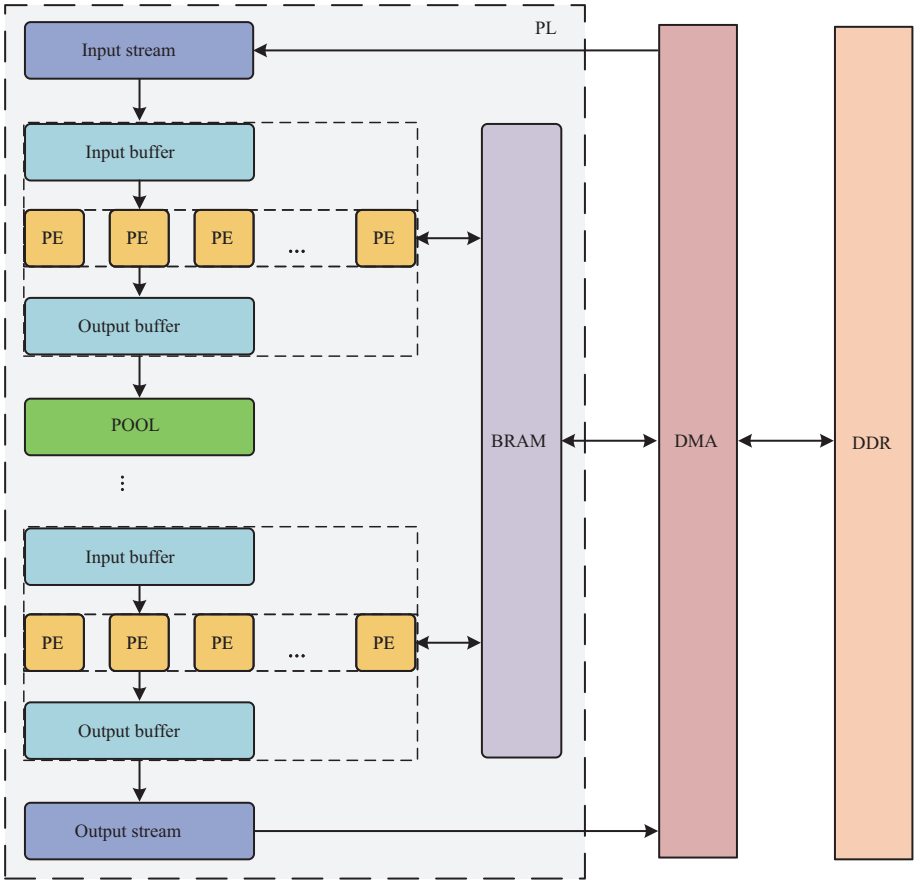

Figure 9 (Color online) Illustration of a streaming accelerator architecture.

Table 4 Comparison of Top-3 FPGA designs in 2020

Entries	BJUT_Runner	Skrskr	iSmart
Network	UltraNet	SkyNet	SkyNet
Quantization	A4/W4	A8/W6	A9/W11
IoU	0.656	0.731	0.724
FPS	212.7	52.4	50.7
GOPS	85.1	48.5	46.9
Energy (J)	1641.2	6764.2	7621.9
Power (mW)	6650	6755	7358
LUT	55.0k (77.5%)	57.6k (81.1%)	58.1k (81.8%)
DSP	360 (100%)	360 (100%)	358 (99.4%)
BRAM (MB)	5.3 (69.7%)	6.9 (90.8%)	7.4 (97.4%)
FF	64.1k (45.5%)	72.2k (51.2%)	64.6k (45.8%)

in 2020, they further compressed the parameters by implementing a 5-bit signed weight quantization scheme instead of a 6-bit signed weight quantization scheme. The weight size was reduced to 2.10 MB from 13.43 MB by quantization. In addition, Skrskr used a DSP packing technique to reduce the use of DSP resources. By computing two multiplications on one DSP (DSP2), 2× parallelism can be obtained with the same number of DSP. Skrskr continued to utilize high-level synthesis for fast FPGA implementation. Different from their design in 2020, they adopted a streaming architecture similar to that of Figure 9 for high hardware performance. As shown in Table 5, the accelerator used 47.8k LUT, 360 DSP, 7.3 MB BRAM, and 39.1k FF resources. It obtained 0.716 IoU, achieved the highest 299.3 FPS, and consumed the least 661.1 J energy.

iSmart [42] adopted UltraNet proposed by BJUT_Runner team in 2020. They adopted a similar

Table 5 Comparison of Top-3 FPGA designs in 2021

Entries	Skrskr	iSmart	SJTU_microe
Network	SkyNet	UltraNet	UltraNet_bypass
Quantization	A8/W5	A4/W4	A4/W4
IoU	0.716	0.708	0.703
FPS	299.3	285.8	249.4
GOPS	277.1	114.3	106.2
Energy (J)	661.1	852.1	1042.3
Power (mW)	3768	4638	4951
LUT	47.8k (67.3%)	46.0k (64.8%)	50.1k (70.6%)
DSP	360 (100%)	357 (99.2%)	306 (85.0%)
BRAM (MB)	7.3 (96.1%)	3.6 (47.4%)	5.1 (67.1%)
FF	39.1k (27.7%)	33.0k (23.4%)	73.8k (52.3%)

quantization scheme of BJUT_Runner team: 4-bit unsigned integer activation in activation layers and 4-bit signed integer weight in all convolutional layers except the first and the last layer. To improve accuracy, they used 8-bit signed integer weights in the first and last layers. IoU improved to 0.708 from 0.656 after retraining. The iSmart team still employed high-level synthesis to facilitate rapid FPGA implementation and adopted a streaming architecture for high hardware performance. Similar to the Skrskr team, iSmart also used the DSP packing technique. However, iSmart implemented a DSP6 design exploiting the convolution feature, which had higher parallelism. As shown in Table 5, the accelerator used 46.0k LUT, 357 DSP, 3.6 MB BRAM, and 33.0k FF resources. Due to the limitation of loading images from the SD card, iSmart did not have a significant advantage in FPS and energy consumption.

SJTU_microe [43] team from Shanghai Jiao Tong University in Shanghai, China proposed an UltraNet_bypass network based on UltraNet proposed by the BJUT_Runner team in 2020 in order to get high accuracy. Different from the original UltraNet, SJTU_microe added an additional bypass approach as shown in Figure 8(b) inspired by SkyNet. The bypass approach could retain more multi-level features for high accuracy, which was first adopted in SkyNet in 2019. The UltraNet_bypass had 381k weights and 213M multiplications. SJTU_microe also adopted the same quantization scheme as BJUT_Runner team, 4-bit weight, and 4-bit activation. The weight size was reduced to 1.45 MB from 11.62 MB by quantization, which was about 2 times of UltraNet. The IoU was improved to 0.703 from 0.656 after retraining. SJTU_microe applied a DSP4 packing design to improve the parallelism of the accelerator. SJTU_microe also employed high-level synthesis to facilitate rapid FPGA implementation. They also adopted the streaming architecture for high hardware performance. As shown in Table 5, the accelerator used 50.1k LUT, 306 DSP, 5.1 MB BRAM, and 73.8k FF resources. It achieved 249.4 FPS and consumed 1042.3 J energy.

In the 2021 LPODC, the DSP Packing technique gained prominence. DSP block (DSP) in the FPGA is an essential and scarce resource for performing these multiplication and accumulation (MAC) operations. Most DSPs have high bit widths, and when used for quantized MACs, most of the bit widths are left underutilized, wasting precious computing resources. Zhu et al. [44] introduced zero gating to minimize energy, allowing DSP to skip the multiplication computation for sparse data flow with zero weights. This method has a low bandwidth requirement and high data sharing. Subsequently, Wang et al. [45] proposed a more efficient accumulator hardware architecture to implement sparse CNNs, transitioning from the original MAC-bound to accumulator-bound. This method achieves a good balance between DSP and logic resources, improving throughput and arithmetic performance. These DSP optimizations are based on the premise that processing sparse data with DSP generates a large number of invalid operations [46]. Optimizing DSP for dense data flow in regular scenarios is more challenging. The DSP Packing technique involves slicing inputs, packing data, and mapping multiple parallelized MACs onto an existing DSP. Over the next two years, it has proven to utilize the computational capacity of DSP units effectively.

3.5 2022's

Out of the 80 participating teams, a total of 26 teams effectively deployed their designs on the provided FPGA platform. The Top-3 teams are SEUer, Ultrateam, and InvolutionNet.

Table 6 Comparison of Top-3 FPGA designs in 2022

Entries	SEUer	Ultrateam	InvolutionNet
Network	UltraNet	UltraNet	UltraNet
Quantization	A4/W4	A4/W4	A4/W4
IoU	0.703	0.703	0.708
FPS	2020.6	2266.2	712.8
GOPS	808.2	906.5	285.1
Energy (J)	36.7	40.3	144.6
Power (mW)	1413	1740	1963
LUT	50.0k (70.4%)	65.7k (92.5%)	52.4k (73.8%)
DSP	345 (95.8%)	360 (100%)	308 (85.6%)
BRAM (MB)	4.8 (63.2%)	5.1 (67.1%)	3.4 (44.7%)
FF	45.0k (31.9%)	48.1k (34.1%)	41.9k (29.7%)

SEUer [17,18,47,48] team from the Southeast University in Nanjing, China adopted UltraNet trained by the iSmart team in 2021 and the no change quantization strategy. Comparing to signed-type DSP packing in 2021, we proposed a more efficient unsigned-type DSP packing, called uint-packing. Our DSP6 implementation based on uint-packing required fewer LUT resources than int-packing. Therefore, we can use more LUT resources to implement multiplications and additions for greater parallelism than iSmart. Eventually, our theoretical system throughput was 1.5× higher than the iSmart team in 2021, with actual testing performance exceeding 2000 FPS, close to the performance boundary of the hardware platform. In addition, we have made some improvements in power optimization, such as introducing multiple clock domains in the block design and dividing the accelerator IP into the high-frequency clock domain to pursue high performance, while all other regions are divided into the low-frequency clock domain to achieve lower power consumption. We adopted a faster method for pre-processing images, which resulted in a little IoU loss (0.005) while the IoU was still larger than 0.7. As shown in Table 6, we achieved the lowest energy consumption in history.

Ultrateam[3)] team from Southeast University in China also adopted the UltraNet trained by the iSmart team in 2021 and achieved high accuracy in object detection. In addition, they adopted a faster pre-processing method for image input, which slightly reduced the IoU by 0.005 but improved the speed significantly. Moreover, they utilized DSP6 packing to enhance the parallelism of computation and optimize the utilization of hardware resources. Comparing to SEUer, another Southeast University team that won the 2020 contest, the Ultrateam team pursued a more aggressive throughput performance strategy by allocating more hardware resources to accelerate computation. As a result, they achieved a theoretical throughput of 2266.2 FPS, the highest record in the contest's history. However, due to the limitation of the FPGA board, its actual improvement in FPS was only about 10% higher than the performance of SEUer, as shown in Table 6.

InvolutionNet[4)] team from the Institute of Computing Technology of the Chinese Academy of Sciences also adopted UltraNet. InvolutionNet used the same quantization strategy and DSP packing as iSmart in 2021, the first layer and the last layer were implemented by DSP2 due to 8-bit quantization, and all other layers were DSP6, so the parallelism was unchanged. In terms of the computational engine, they modified the original computational template and redesigned a structure called serialACT. The serialACT was based on the principle of moving the original functions that compute Batch Normalization and activation out of the convolutional computation module and working as a separate module for pipelining, which could save a small amount of DSP resources, but not enough to improve the overall parallelism. Furthermore, their image preprocessing speed was not as fast as that of SEUer and Ultrateam. Therefore, their FPS was only about 1/3 of SEUer and Ultrateam while consuming 3× energy.

The streaming structure has shown significant advantages since its introduction in 2020. In 2021 and 2022, the Top-3 teams were exclusively based on the streaming structure. Among them, SEUer and Ultrateam emerged as exemplars in the streaming structure, fully leveraging the parallel advantages of the pipeline. They approached the theoretical performance limits of the Ultra96 hardware platform.

3) Dac-sdc-2022-ultrateam. https://github.com/jgoeders/dac_sdc_2022_designs/tree/main/ultrateam.

4) Dac-sdc-2022-involutionnet. https://github.com/jgoeders/dac_sdc_2022_designs/tree/main/InvolutionNet.

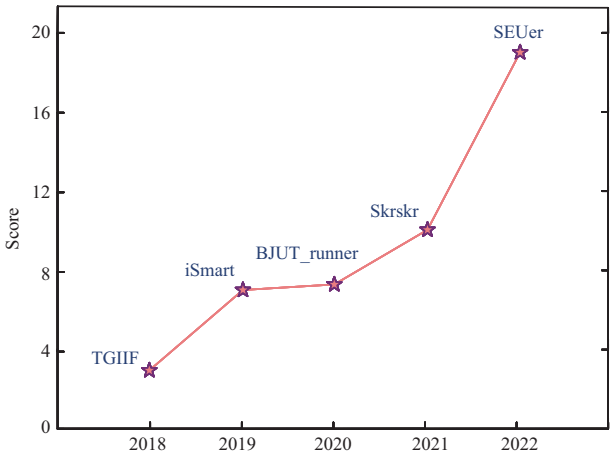

Figure 10 (Color online) Total score of the champion team for each year from 2018 to 2022 according to the evaluation method of 2022.

However, the complete streaming structure only began to emerge two years after its introduction in LPODC, indicating substantial deployment challenges. Finally, summarize the strengths and weaknesses of the streaming structure. In terms of strengths, all data follow a pipelined path through the CEs of each layer, keeping each CE essentially active and enhancing computational efficiency. Additionally, no storage of intermediate feature maps or access to external memory is required, achieving maximum throughput. When considering weaknesses, the deployment threshold for the streaming structure is high. First, all weights must be quantized and stored in on-chip memory, imposing high requirements on quantization algorithms and on-chip storage capacity. To adapt to the limited storage capacity on the chip, more aggressive quantization schemes are necessary, leading to significant quantization losses. Furthermore, the depth and parameter count of the network also constrain the feasibility of the final deployment.

4 Results and analysis

4.1 Total score

In 2018, the hardware platform was PYNQ-Z1 with a 28 nm process, while from 2019 to 2022, the hardware platform was the Ultra96 series with a 16 nm process. To objectively evaluate each design, we follow the official datasheet of Xilinx [49][5)], where the Normalized Total Power of UltraScale+ (16 nm) is 0.8x that of the ZYNQ7000 series (28 nm). For all designs in 2018, we made corresponding power compensations. Additionally, due to the difference in the total DSP of core computing resources between PYNQ-Z1 and Ultra96, we made inference speed compensations for all designs based on PYNQ-Z1 according to the ratio of the corresponding DSP resource totals. The evaluation methods were different from 2018 to 2022. We ultimately recalculated the total score of each year's champion team from 2018 to 2022 according to the 2022 evaluation method.

As shown in Figure 10, the total score increases year by year. The key to TGIIF winning the 2018 championship was that the network they proposed achieved the highest IoU of that year. However, despite normalization, there was a significant gap from the desired 30 fps, resulting in a substantial penalty on the FPS metric and consequently leading to a low overall score. In 2019, iSmart won the championship. The keys were the excellent network SkyNet (IoU>0.7 at the first time) and the algorithm-hardware co-design, which was adopted by many teams in 2020 and 2021. In 2020, BJUT_runner obtained the championship. The keys were the streaming architecture accelerator and the LSFQ quantization method, which helped them achieve fast FPS and low energy consumption. The streaming architecture and the LSFQ quantization method were widely adopted by participating teams in 2021 and 2022. However, the precision penalty incurred due to an IoU below 0.7 resulted in a limited overall score improvement compared to the iSmart benchmark of the previous year. In 2021, Skrskr obtained the SkyNet-based championship. The keys were the TAIT quantization method, fine-grained multi-threading, and high

5) Xilinx-power-efficiency. https://www.xilinx.com/products/technology/power.html.

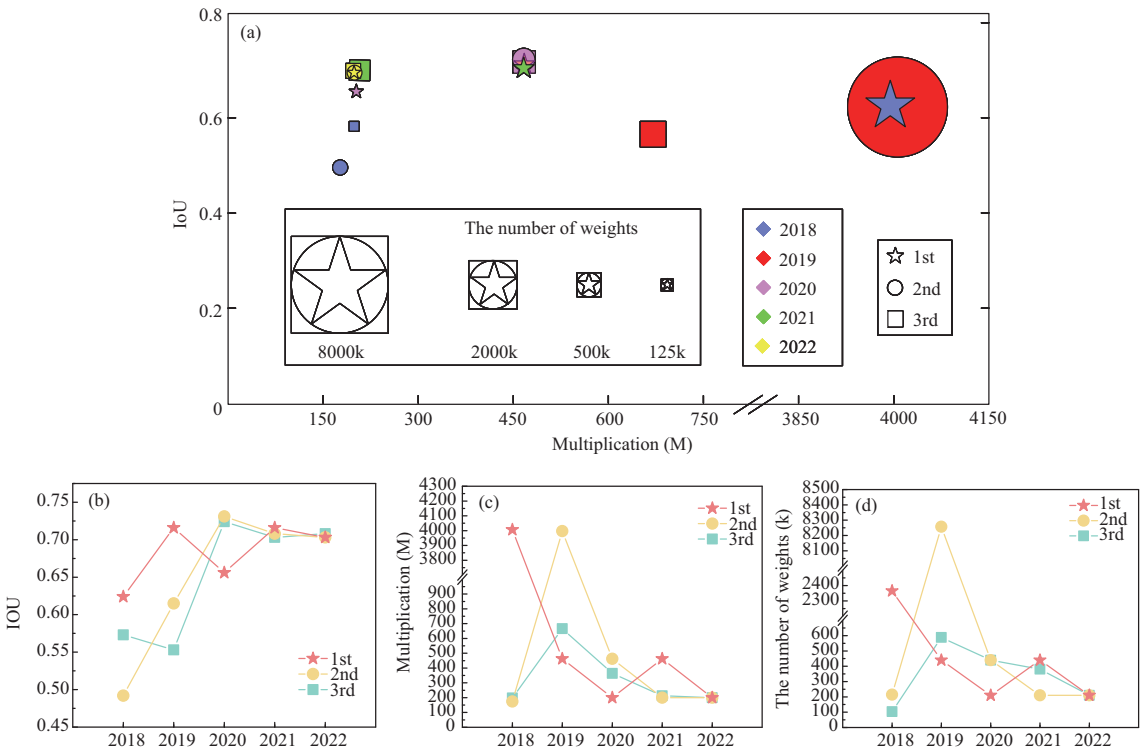

Figure 11 (Color online) (a) IoU, the number of weights and the number of multiplications of the top three teams of each year from 2018 to 2022. (b) Temporal trend of IoU, (c) the number of multiplications, and (d) the number of weights metric from 2018 to 2022.

parallelism by DSP2 packing. In 2022, our SEUer team obtained the championship, The keys were uint-packing DSP6 high parallelism, multiple clock domains for low power, and streaming architecture. In 2022, the score increase was very high because the runtime was only the PL's working time rather than the whole system's working time for avoiding the limitation of image loading. SEUer and Ultrateam almost achieved the theoretical limit performance on the Ultra96 V2 board.

4.2 Network and IoU

The number of weights, IoU, and multiplications of the top three teams in each year from 2018 to 2022 are shown in Figure 11. In 2018, the IoU of Top-3 designs was lower than 0.7. The championship TGIIF adopted a large network and obtained the champion at 0.624 IoU. It should be mentioned that the 2nd place SystemsETHZ adopted a binary network and obtained 0.492 IoU. The 3rd place adopted lightweight DSC. In 2018, the entries adopted a variety of convolutions and networks. In 2019, the iSmart2 championship proposed the construction of SkyNet by separable convolution in depth, whose IoU was higher than 0.7 for the first time. The 2nd place XJTU_Tripler adopted a very large network. In 2018 and 2019, the accuracy was very important and the team with the highest IoU would win the championship. In 2020, Top-3 teams adopted small networks. The championship BJUT_Runner proposed a smaller UltraNet than SkyNet and achieved 0.656 IoU. Both the 2nd place team and the 3rd place team adopted SkyNet, because it was very difficult to design a network with IoU larger than 0.7. From 2020, all Top-3 participant entries adopted UltraNet or SkyNet and began to work to reduce energy consumption. In 2021, the iSmart team re-trained UltraNet and improved IoU to 0.708 from 0.656. iSmart team was very good at training networks, almost all networks with IoU larger than 0.7 were trained by them. In 2022, the Top-3 teams all used UltraNet proposed by BJUT_Runner, and trained by iSmart. There were no improvements in detection accuracy achieved and the winning strategy was to shift toward hardware efficiency.

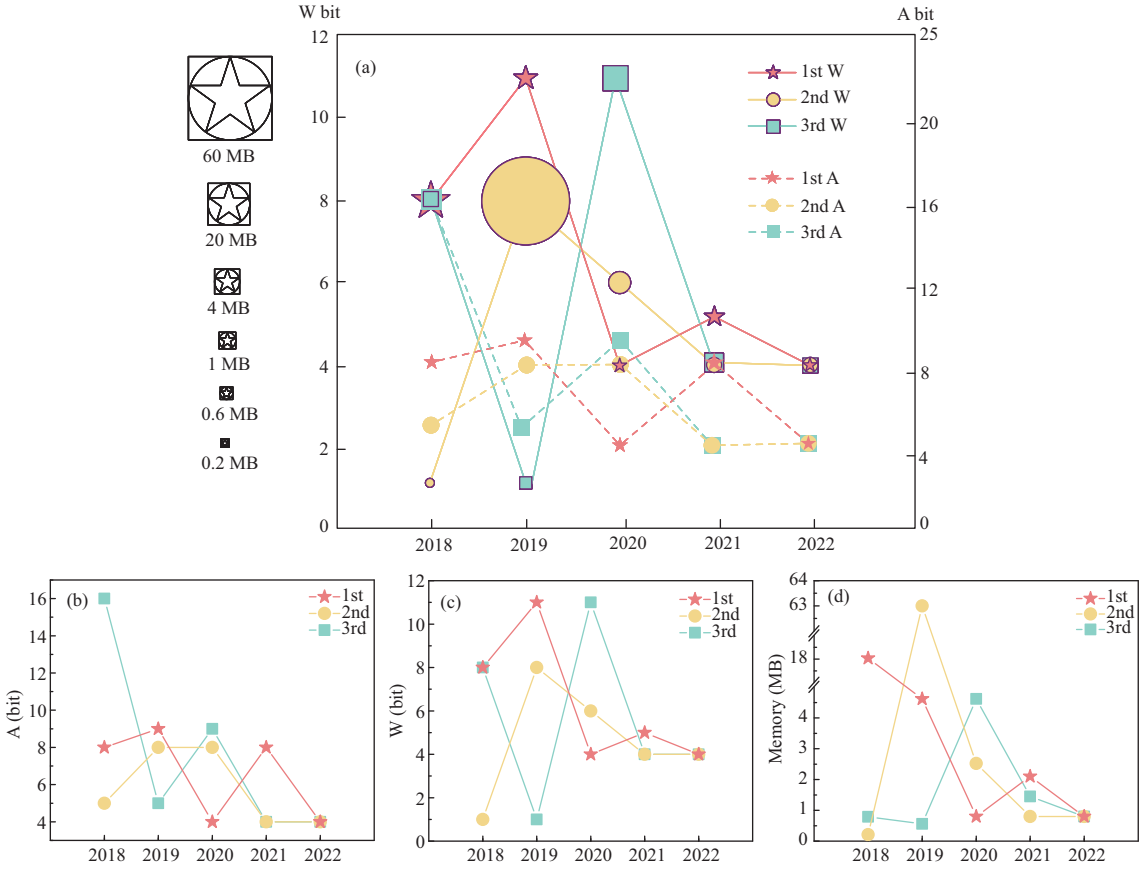

Figure 12 (Color online) (a) Quantization approaches of the top three teams each year from 2018 to 2022. (b) Temporal trend of activation quantization bit width, (c) weight quantization bit width, and (d) memory occupation from 2018 to 2022.

4.3 Quantization

Quantization plays a pivotal role in deploying CNNs on the FPGA platform. The full precision floating-point weight and activation can be quantized to be a low-bit fixed point or integer due to the redundancy of CNN. Quantization plays a pivotal role in significantly reducing the dimensions of weights and intermediate activation parameters during network inference, leading to a marked decrease in memory occupation and computational intensity. It is worth noting that quantization is intricately linked to the development of specialized architectures that efficiently execute network inference by mapping them onto low energy and low-bit integer digital circuits. The quantization approaches of the top three teams in each year from 2018 to 2022 are shown in Figure 12. For weights, the size of the symbol represents the weight size after quantization. In 2018 and 2019, SystemsETHZ adopted binary weight, so the weight size was very small and the other weight sizes were large. The binary weight was deprecated for high accuracy after 2019 because using binary weight was difficult to achieve high accuracy. Since 2020, the quantization method was beginning to gain traction. In 2020, the championship BJUT_Runner introduced a novel approach known as learnable parameter soft clipping full integer quantization (LSFQ). This technique encompasses the quantization of both weights and activations, with the incorporation of learnable clipping parameters. In 2021, the Skrskr championship adopted tunable activation imbalance transfer (TAIT) for quantization based on SkyNet. Once again, SkyNet was adopted by the participating team to win the championship due to the advanced quantization method. In 2022, the Top-3 teams adopted the LSFQ based on UltraNet. For activation parameters, almost all teams adopted 4-8bit quantization strategies between 2019 and 2022.

4.4 Throughput and energy consumption

LPODC evaluates the latency of all designs using the FPS metric. Considering that DSP resources are the primary computing resources in FPGA, and the total DSP resources differ between hardware platforms

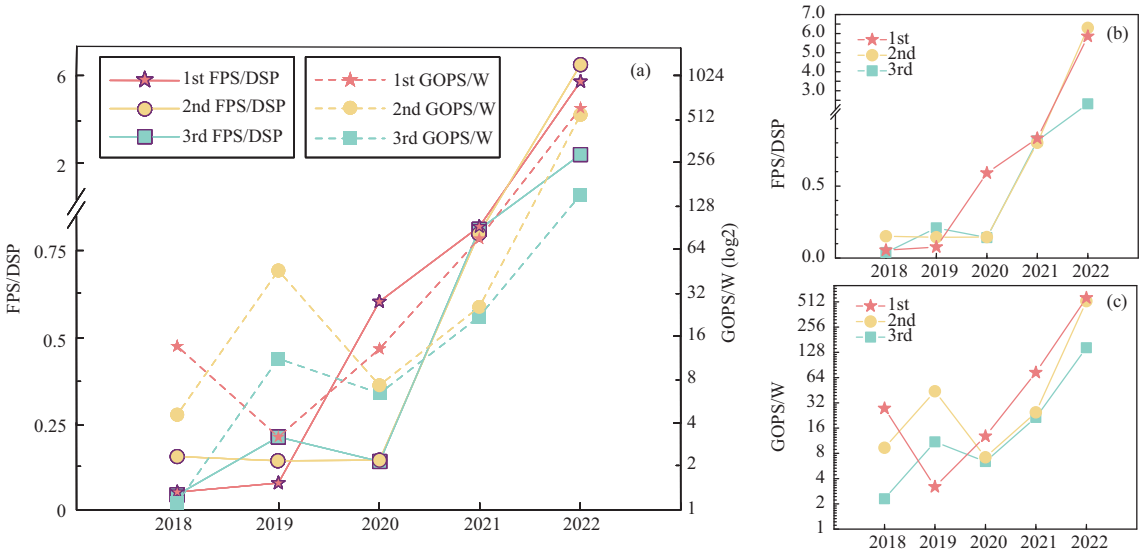

Figure 13 (Color online) (a) FPS/DSP and GOPS/W of the top three teams in each year from 2018 to 2022. Temporal trend of (b) FPS/DSP and (c) GOPS/W metric from 2018 to 2022.

(360 for Ultra96 series and 220 for PYNQ-Z1), along with variations in DSP usage across different designs, normalizing the FPS metric to the actual DSP usage in each design (FPS/DSP) can more scientifically assess the computational resource efficiency during runtime. It reflects the contribution of DSP to accelerating computations and reducing latency during operation. Regarding power consumption, the corrected GOPS/W, considering the impact of the manufacturing process on power consumption, reflects the number of billion operations that a hardware accelerator can perform per watt. Thus, a higher GOPS/W indicates that the hardware accelerator can provide more computational performance per unit power, demonstrating better energy efficiency. This is particularly crucial in environments with power constraints, such as many embedded systems and mobile devices. Compared to the original energy (J) metric of LPODC, the GOPS/W metric reflects efficiency and helps strike a balance between performance and power consumption.

The FPS/DSP and GOPS/W of the top three teams in each year from 2018 to 2022 are shown in Figure 13. In 2018, the FPS and FPS/DSP were very low, the 2nd place SystemsETHZ achieved the highest FPS using binary weight. The optimal energy efficiency was achieved by the championship TGIIF, which can be attributed to its optimized structure of the systolic array. In 2019, the Top-3 teams achieved higher throughput and energy efficiency. SystemsETHZ achieved the highest FPS and FPS/DSP due to the more efficient binary weight. The best energy efficiency was achieved by the 2nd place XJTU_Tripler, as it specifically designed an MPU unit for matrix multiplication. In 2020, BJUT_Runner used the LSFQ quantization method on UltraNet, as well as the streaming architecture accelerator. All weights were stored in on-chip memory, preventing memory access to the weights during the inference phase. The accelerator comprised a dedicated processing engine for each layer, with inputs seamlessly flowing through the dataflow architecture, allowing for parallel computation across all layers. The intermediate features were streaming in the accelerator, which avoided memory access for the feature map during the inference phase. The FPS of BJUT_Runner exceeded 200 and the FPS/DSP ratio also exceeded 0.5 for the first time. Hardware efficiency improved greatly in 2020. In 2021, the championship Skrskr further inspired the hardware's potential by implementing fine-grained multi-threading, streaming architecture, and high parallelism by DSP packing. By computing two multiplications on one DSP (DSP2), half of the DSP can be saved with the same parallelism. Its energy efficiency ratio reached a new high in the past four years, exceeding 64 GOPS/W. In 2022, FPS and energy efficiency improved significantly. SEUer team adopted the streaming architecture, a more efficient DSP6 uint-packing method, and higher parallelism. SEUer achieved the best energy efficiency ratio in the history of LPODC and more than 2000 high FPS.

4.5 Summary and recommendations

We look forward to a more in-depth analysis of the LPODC results, which can provide additional insights for object detection tasks in different application scenarios. In general, for high-accuracy scenarios, people

tend to favor larger and more complex models and may employ more conservative quantization schemes. On the contrary, a simple, parameter-efficient model coupled with an aggressive quantization scheme can exhibit outstanding performance in low-power scenarios. The choice of design strategy often varies significantly for different application scenarios. If the focus is solely on improving algorithm precision without considering hardware deployment challenges, it may result in poor inference performance and increased system power consumption. Similarly, highlighting hardware inference speed without regard for object detection recognition accuracy can lead to substantial quantization losses and identification errors. Therefore, whether in high-accuracy or low-power scenarios, the concept of algorithm-hardware co-design is essential. In high-accuracy scenarios, precision requirements are generally high, while power consumption and inference speed requirements are low. Therefore, during co-design, there is a higher tolerance for parameter and computation volume, allowing for special model structure designs (such as the FPN pyramid structure) to enhance detection accuracy. However, in low-power scenarios, a simple model structure with a large number of repetitions (such as the VGG straight-through structure) is convenient for pipeline design, significantly boosting parallel computing capabilities. This setup achieves high throughput while maintaining excellent power efficiency.

Through comprehensive analysis, we observed that due to the restricted DSP input bit-width, DSP4 and even DSP6 solutions are based on 4-bit quantization schemes. When the quantization bit-width exceeds 4 bits, the optimal DSP parallel solution becomes the DSP2 scheme. Therefore, in high-accuracy scenarios, the optimal quantization scheme is W8/A8. This not only achieves the best quantization performance but also enables the hardware architecture design based on the DSP2, enhancing computing power and reducing total power consumption. In low-power scenarios, W4/A4 is the optimal choice. If the quantization scheme is more aggressive, it will lead to a sharp increase in quantization error, and at this point, the DSP6 computing architecture can already be implemented, and further reducing quantization does not yield significant benefits. It is worth mentioning that, in testing, we found that compared to weights, activations play a more significant role in preserving accuracy. However, the minimum weight cannot be lower than 3 bits. Therefore, in scenarios that demand low power consumption while also requiring high accuracy, it may be worthwhile to consider W3/A5. This approach retains the ability to use DSP6 and achieves a higher recognition accuracy than W4/A4.

5 Conclusion

LPODC, hosted at SDC-DAC, has witnessed significant success over the past five years, and we anticipate its continued prominence as a premier contest in the fields of low-power object detection and algorithm-hardware co-design for AI. In this paper, we present the LPODC for the UAV object detection domain, including dataset, hardware platform, and evaluation method. Furthermore, we introduce and delve into the methodologies proposed by the Top-3 teams each year from 2018 to 2022. Specifically, we analyze the solutions of each year's Top-3 teams in terms of network, accuracy, quantization method, hardware performance, and total score. This analysis of the results serves as a valuable practical resource for researchers and engineers.

LPODC serves as an initial stepping stone within the realm of TinyML, a field dedicated to the deployment of AI on compact hardware platforms featuring limited resources. There will be an increasing number of TinyML contests in the future, such as the TinyML contest at ICCAD 2022. We obtained the 2nd place in TinyML at ICCAD2022. The achievements of the LPODC winners underscore the significance of adopting a hardware-algorithm co-design approach within the realm of TinyML. This approach prioritizes the integration of hardware and algorithm optimization rather than treating them as separate entities. Furthermore, it is anticipated that future competitions may expand to encompass areas such as compiler and device optimization. These advancements are poised to propel the field of TinyML, ultimately making it accessible, reliable, and widely applicable in daily scenarios.

Acknowledgements This work was supported by Key R&D Program of Guangdong Province (Grant No. 2021B1101270006), Shandong Provincial Natural Science Foundation (Grant No. ZR2023QF056), and Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. SJCX22_0051).

References

1 Zhang Q, Zhang M, Chen T, et al. Recent advances in convolutional neural network acceleration. Neurocomputing, 2019, 323: 37–51

2 Li G, Zhang M, Wang J, et al. SCWC: structured channel weight sharing to compress convolutional neural networks. Inf Sci, 2022, 587: 82–96

3 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: Proceedings of International Conference on Learning Representations, 2015

4 He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016. 770–778

5 He K, Zhang X, Ren S, et al. Identity mappings in deep residual networks. In: Proceedings of European Conference on Computer Vision, 2016. 630–645

6 Huang G, Liu Z, van der Maaten L, et al. Densely connected convolutional networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2017. 2261–2269

7 Li G, Zhang M, Li J, et al. Efficient densely connected convolutional neural networks. Pattern Recogn, 2021, 109: 107610

8 Liu B, Zhang Z, Cai H, et al. Self-compensation tensor multiplication unit for adaptive approximate computing in low-power CNN processing. Sci China Inf Sci, 2022, 65: 149403

9 Zhang Z, Chen J, Chen X, et al. From macro to microarchitecture: reviews and trends of SRAM-based compute-in-memory circuits. Sci China Inf Sci, 2023, 66: 200403

10 Li G, Shen X, Li J, et al. Diagonal-kernel convolutional neural networks for image classification. Digital Signal Process, 2021, 108: 102898

11 Li G, Zhang M, Zhang Y, et al. Efficient channel expansion and pyramid depthwise-pointwise-depthwise neural networks. Appl Intell, 2022, 52: 12860–12872

12 Li G, Zhang M, Zhang Q, et al. Efficient binary 3D convolutional neural network and hardware accelerator. J Real-Time Image Proc, 2022, 19: 61–71

13 Li G, Zhang M, Zhang J, et al. OGCNet: overlapped group convolution for deep convolutional neural networks. Knowledge-Based Syst, 2022, 253: 109571

14 Shan W, Cui Y, Dai W, et al. An efficient path delay variability model for wide-voltage-range digital circuits. Sci China Inf Sci, 2023, 66: 129401

15 Xu X, Zhang X, Yu B, et al. DAC-SDC low power object detection challenge for UAV applications. IEEE Trans Pattern Anal Mach Intell, 2021, 43: 392–403

16 Jia Z, Xu X, Hu J, et al. Low-power object-detection challenge on unmanned aerial vehicles. Nat Mach Intell, 2022, 4: 1265–1266

17 Li G, Zhang J, Zhang M, et al. An efficient FPGA implementation for real-time and low-power UAV object detection. In: Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS), 2022. 1387–1391

18 Li G, Zhang J, Zhang M, et al. Efficient depthwise separable convolution accelerator for classification and UAV object detection. Neurocomputing, 2022, 490: 1–16

19 Torres-Sánchez J, López-Granados F, Peña J M. An automatic object-based method for optimal thresholding in UAV images: application for vegetation detection in herbaceous crops. Comput Electron Agr, 2015, 114: 43–52

20 Wu X, Li W, Hong D, et al. Deep learning for unmanned aerial vehicle-based object detection and tracking: a survey. IEEE Geosci Remote Sens Mag, 2022, 10: 91–124

21 Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. Int J Comput Vis, 2015, 115: 211–252

22 Everingham M, Van Gool L, Williams C K I, et al. The pascal visual object classes (VOC) challenge. Int J Comput Vis, 2010, 88: 303–338

23 Zeng S, Chen W, Huang T, et al. DAC2018-TGIIT. 2018. https://github.com/hirayaku/DAC2018-TGIIF

24 Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector. In: Proceedings of European Conference on Computer Vision, 2016. 21–37

25 Zeng S, Kara K, Zhang C, et al. DAC2018-systemsETHZ. 2018. https://github.com/fpgasystems/spooNN

26 Iandola F N, Moskewicz M W, Ashraf K, et al. Squeezenet: alexnet-level accuracy with 50x fewer parameters and $<$ 0.5 MB model size. 2016. ArXiv:1602.07360

27 Redmon J, Divvala S K, Girshick R B, et al. You only look once: unified, real-time object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016. 779–788

28 Qin H, Gong R, Liu X, et al. Binary neural networks: a survey. Pattern Recogn, 2020, 105: 107281

29 Hao C, Li Y, Huang S, et al. DAC2018-iSmartDNN. 2018. https://github.com/onioncc/iSmartDNN

30 Howard A G, Zhu M, Chen B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications. 2017. ArXiv:1704.04861

31 Zhang X, Lu H, Hao C, et al. Skynet: a hardware-efficient method for object detection and tracking on embedded systems. In: Proceedings of Machine Learning and Systems, 2020

32 Zhang X, Hao C, Li Y, et al. A Bi-directional Co-design approach to enable deep learning on IoT devices. 2019. ArXiv:1905.08369

33 Hao C, Zhang X, Li Y, et al. FPGA/DNN Co-design: an efficient design methodology for IoT intelligence on the edge. In: Proceedings of the 56th ACM/IEEE Design Automation Conference (DAC), 2019. 1–6

34 Zhang X, Hao C, Lu H, et al. SkyNet: a champion model for dac-sdc on low power object detection. 2019. ArXiv:1906.10327

35 Zhao B, Xia T, Zhai H, et al. REMAP: a spatiotemporal CNN accelerator optimization methodology and toolkit thereof. IEEE Trans Comput-Aided Des Integr Circ Syst, 2023, 42: 1691–1704

36 Zhao C, Zhao W, Xia T, et al. DAC2019-XJTU-Tripler. 2019. https://github.com/xjtuiair-cag/XJTU-Tripler

37 Bao Z, Guo J, Li X, et al. MSCU: accelerating CNN inference with multiple sizes of compute unit on FPGAs. In: Proceedings of IEEE International Symposium on Embedded Multicore/Many-core Systems-on-Chip, 2021. 106–113

38 Bao Z, Fu G, Zhang W, et al. LSFQ: a low-bit full integer quantization for high-performance FPGA-based CNN acceleration. IEEE Micro, 2022, 42: 8–15

39 Bao Z, Zhan K, Zhang W, et al. LSFQ: a low precision full integer quantization for high-performance fpga-based CNN acceleration. In: Proceedings of IEEE Symposium in Low-Power and High-Speed Chips, 2021. 1–6

40 Jiang W, Yu H, Liu X, et al. TAIT: one-shot full-integer lightweight DNN quantization via tunable activation imbalance transfer. In: Proceedings of the 58th ACM/IEEE Design Automation Conference (DAC), 2021. 1027–1032

41 Chen S, Zhou Z, Ha Y. An ultra energy efficient streaming-based FPGA accelerator for lightweight neural network. In: Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS), 2022. 3111–3114

42 Liu X, Chen Y, Ganesh P, et al. HiKonv: high throughput quantized convolution with novel bit-wise management and computation. In: Proceedings of the 27th Asia and South Pacific Design Automation Conference (ASP-DAC), 2022. 140–146

43 Du P, Deng G, Kong Y, et al. Dac2021-sjtu_microe. https://github.com/heymesut/SJTU_microe

44 Zhu C, Huang K, Yang S, et al. An efficient hardware accelerator for structured sparse convolutional neural networks on FPGAs. IEEE Trans VLSI Syst, 2020, 28: 1953–1965

45 Wang D, Xu K, Guo J, et al. DSP-efficient hardware acceleration of convolutional neural network inference on FPGAs. IEEE Trans Comput-Aided Des Integr Circ Syst, 2020, 39: 4867–4880

46 Wu W, Tu F, Li X, et al. SWG: an architecture for sparse weight gradient computation. Sci China Inf Sci, 2024, 67: 122405

47 Zhang J, Cao X, Zhang Y, et al. Dac-sdc-2022-seuer. https://github.com/AiArtisan/dac_sdc_2022_champion

48 Zhang J, Zhang M, Cao X, et al. Uint-packing: multiply your dnn accelerator performance via unsigned integer DSP packing. In: Proceedings of the 60th ACM/IEEE Design Automation Conference (DAC), 2023. 1–6

49 Steiner G, Philofsky B. Managing power and performance with the Zynq UltraScale+ MPSoC. In: Proceedings of White Paper: Zynq UltraScale+ MPSoC, WP482 (v1.1), 2016