# Chapter 1
# Introduction

## 1.1 Research Background and Significance

### *1.1.1 Development Trends of Neural Network*

Artificial Intelligence (AI) has promoted the development of modern society in many aspects [1], and will profoundly change human social life and the world [2]. Artificial intelligence has been applied in the fields including social security, medical treatment, finance, traffic navigation, agriculture, etc., which shows an increasing number in consumer products. In 2017–2018, a number of governments, including China and the United States, have released the development strategies of artificial intelligence [3]. It can be observed that artificial intelligence owns a broad prospect. The relevant research is of considerable significance to national strategy and social development.

Neural Network (NN) is a typical algorithm implementation of the modern artificial intelligence. With the development of big data, adequate computing resources, and advanced NN algorithm model, the fast developing neural network technique presents much better results than the traditional algorithms in recent years. In the 2012 ImageNet Large Scale Visual Recognition Challenge [4], the multi-layer convolutional neural network (CNN) algorithm improved the Top-5 accuracy to 84.7% [5], which is 10.8% higher compared with the conventional algorithms with manually designed features. The following several works further improve the Top-5 accuracy to 96.4% [6–9]. The classification accuracy of neural networks has exceeded humans since 2014.

The NN algorithm models are developing rapidly, and the application field is expanding. NN models were used in the field of computer vision in the early stage, such as image classification, recognition, etc. One representative algorithm model is the convolutional neural network (CNN) [5–9], including convolutional layers and fully connected layers.

The recurrent neural network (RNN), e.g. long short-term memory (LSTM), shows good performance in the recognition of speech, text and other sequence information, which can be widely applied on automatic translation, image to text conversion, voice to text conversion, etc. At present, the self-attention NN model [10] (transformer) is also widely used in the field of natural language processing. Besides, the generative adversarial network [11] (GAN) is applied to image generation, such as super-resolution image restoration, image style migration, semantic segmentation, etc. The AlphaGo intelligent algorithm [12] developed by Google has defeated world Go Champions Sedol Lee and Jie Ke in 2016 and 2017, respectively. Neural network is also gradually applied to a wider range of practical scenarios, such as web page recommendation, financial risk prediction, vaccine research and development, chip design optimization, navigation, weather forecasting, and so on.

In summary, neural network and its application have been widely applied to almost every aspect of human society. AI chip, i.e. NN processor, is the hardware foundation to implement neural network algorithm. A research report from Askci Consulting company predicts that the global artificial intelligence chip market will exceed 70 billion US dollars by 2025 [13]. The research of NN processor will promote the practical application of NN algorithm in various intelligent devices, which further promotes social development and progress.

### *1.1.2 Requirements of NN Processor*

Neural network algorithm requires a huge amount of computation and parameters. Figure 1.1 in previous work [14] has shown the accuracy and complexity of various neural network models on the ImagenNet dataset. The horizontal and vertical coordinates are the number of floating-point operations required to perform one NN inference operation and the Top-1 accuracy of the NN model, respectively. The size of each circular icon represents the number of parameters required by the NN model. With the increasing demand for accuracy, the amount of computation and the number of parameters show an upward trend. These NN models generally require $10^9$–$10^{10}$ computation and $10^6$–$10^8$ parameters. Such a huge amount of computation, storage, and memory access overhead will seriously affect the performance and power consumption of NN processors. In addition, the parameters of NN structures are also diverse. Supporting different NN parameters will also affect the performance and power consumption of NN processor to a certain extent.

Conventional general-purpose processor (GPP) cannot satisfy the need for a wide range of low-power intelligent devices to run NN algorithms. Central Processing Unit (CPU) is the most common GPP solution at the early stage. However, as the size of NN models grows, CPU cannot meet the training and inference requirement for large-scale NN models. For example, a VGG model of the ImageNet dataset requires 818 ms inference time [15] for each image on the Qualcomm Snapdragon 855 CPU, which can hardly satisfy the real-time requirements. For CPU platforms with lower power supply, such as the ARM Cortex-M0, the processing time of NN model may reach several seconds or even hundreds of seconds.

Graphics Processing Unit (GPU) provides sufficient computational support for large-scale training and reasoning of NN models. However, GPU cannot meet the

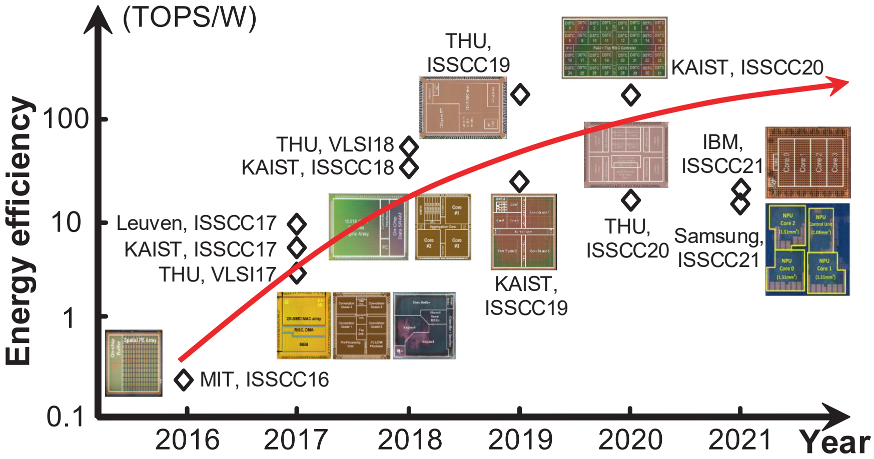

**Fig. 1.1** Related research on digital NN processors

low-power requirement. GPU owns a large number of parallel computing units. The latest NVIDIA A100 GPU can reach the peak performance of 624TFLOPS [16].[1]. However, the high performance of GPU comes at the cost of high power consumption. Even for the mobile GPU [17], it still requires 5–10 W power consumption. As a general-purpose high-parallelism processor, GPU lacks special optimization for NN algorithms, and a large number of GPU computing resources cannot be fully utilized.

Field Programmable Gate Array (FPGA) accelerated the early-stage development, iteration, and deployment of NN processors due to its high flexibility and fast reconfiguration. FPGA is a programmable hardware processor, which can quickly convert hardware description language (HDL) into corresponding hardware implementation. In the period of rapid development and change of NN models, FPGA can quickly adjust the hardware architecture according to the change of algorithms to realize rapid iteration and deployment. As the stability of NN model and the deepening of NN processor optimization, the inherent characteristics of FPGA restrict its further reduction of power consumption and energy efficiency improvement. On one hand, compared with the fully customized application specific integrated circuits (ASIC) design, the semi-customized programmable gate array adopted by FPGA brings a larger area and power overhead. On the other hand, some specifical circuit designs are difficult to be implemented on FPGA, and must be implemented by ASIC chips.

In conclusion, traditional general-purpose processors (CPU, GPU and FPGA) are difficult to meet the needs of a wide range of low-power intelligent devices in terms of performance and power consumption. The energy-efficient specific processor for NN algorithm is required, which is the core reserach top of this book. A report from the Chinese government points out that “Aim at artificial intelligence, quantum information, integrated circuits, life and health ... Implement a number of forward-

[1] The concepts and calculation methods of performance and OPS are detailed in Sect. 2.2

looking and strategic national science and technology projects" [18]. The research on energy-efficient NN processor is oriented to two important scientific fields (artificial intelligence and integrated circuit), which is of important strategic significance and social value.

### *1.1.3 Energy-Efficient NN Processors*

The existing research work of NN processors can be mainly divided into two categories: conventional digital NN processors and computing-in-memory (CIM) NN processors. In the digital NN processors, all the storage and computation of the NN algorithm are realized by the digital circuits, which are featured with separated storage and calculation unit. Different from that, the CIM NN processor combines the storage unit and the computing unit together, and completes the computing operation inside the storage unit.

The digital NN processor has adopted three main techniques: data reuse, low-bit quantization and NN model compression. The representative research work of digital NN chips is shown in Fig. 1.1. The early-stage work of NN processors [19–21] adopts the data reuse characteristics of NN algorithms. By optimizing the chip architecture and data path, it can reduce the memory access at each level, especially the bottom-level memory access, to reduce power consumption. With the in-depth study of NN algorithms, the technical route of reducing algorithm redundancy through low bit quantization [22, 23] and NN model compression [24–28] is gradually adopted. In the aspect of low bit quantization, many kinds of circuit optimization techniques, such as time-domain and space-domain multiplication splitting, are adopted [29, 30]. The initial floating-point operations are transferred to low bit fixed-point calculation to reduce the area and power consumption of storage and computation. NN model compression, represented by sparse pruning technique, is applied to accelerate NN processors [31–34]. By compressing zero-value storage and skipping zero-value calculation, computation time and power consumption are saved to achieve equivalent energy efficiency improvement.

Computing-in-memory (CIM) architecture implements storage and computation in the same circuit unit. In modern NN processor architecture, the proportion of memory access power is more important than computation power consumption [35]. The concept of Computing-in-memory has emerged in the 1990s and has been used to implement specific circuit functions [36–39]. In recent years, CIM technology based on regular array structure has been applied to the architecture and chip design of NN processors [40–50]. The representative research work is shown in Fig. 1.2.

Compared with the digital NN processor based on von Neumann architecture, CIM architecture integrates memory and computing array together, reducing the cost of reading a large amount of data from memory. The Related work has been explored from the level of bottom-level devices, circuits, and CIM intellectual property (IP) cores. At device level, conventional storage devices, such as the static random access memory (SRAM), dynamic random access memory (DRAM) and Flash, are adopted

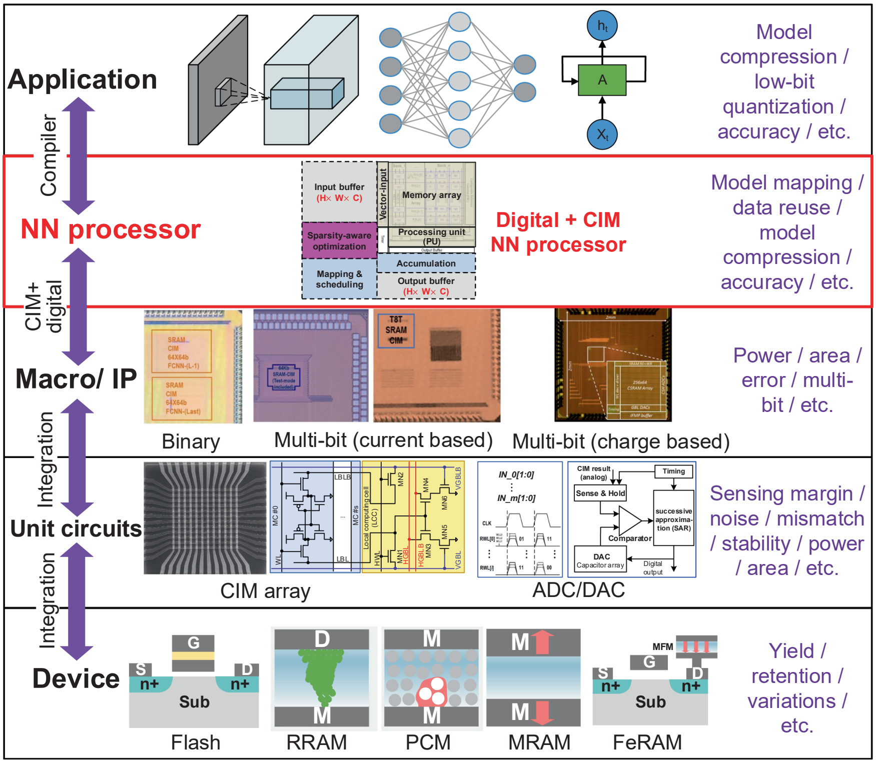

**Fig. 1.2** Related research on CIM NN processors

in CIM circuits [45, 46, 51–54]. Several emerging non-volatile memory (NVM) devices such as the resistive random access memory (RRAM), ferroelectric random access memory (FeRAM), phase change memory (PCM), and magnetic random access memory (MRAM), are also considered for CIM circuits [43, 55–58]. The emerging NVM devices show advantages in read/write power, density, etc. compared with the conventional storage devices, but there is still a certain gap in reliability, consistency and process maturity compared with traditional memory devices. At the circuits level, related research has explored digital-to-analog converter (DAC), CIM array, analog-to-digital converter (ADC), etc. [40–49], which aims to improve stability and accuracy, reducing errors, reducing power/area, etc. At CIM IP core level, there have been several IP-level CIM chips utilizing CIM architecture [40–49], which verifies the feasibility and high energy efficiency advantage of implementing NN algorithm on CIM architectures.

Previous related research has explored the design of NN processors in two ways: digital circuits and computing-in-memory, which effectively reduces the power consumption and area overhead, and improve the energy efficiency of NN processor. However, there are still unexplored scientific problems and still much room for optimization. In the digital circuit NN processor, the utilization of data reuse is insufficient. A part of data still needs to be accessed repeatedly, which still requires large area overhead or memory access power consumption. Irregular sparsity optimization technology leads to significant additional power/area overhead, which limits the further improvement of energy efficiency. In the CIM NN processors, there is a lack of research on the system-level CIM chips. A single macro-level CIM IP cannot constitute a complete CIM system. At present, there is little research on network mapping, data reuse and sparse optimization in CIM chips. It also lacks system-level optimization to integrate CIM IPs and digital circuits. In addition, there is still a gap between the current small-scale on-chip CIM IP capacity and the large-scale practical NN models. It is necessary to explore the optimization of energy-efficient CIM chip to support the large-scale NN models.

## 1.2 Summary of the Research Work

### *1.2.1 Overall Framework of the Research Work*

To sum up, the energy-efficient application-specific processor is the foundation and core component of low-power intelligent devices. However, there is a large optimization space for the existing NN processors, which needs to be further optimized to achieve higher energy efficiency.

The theme of this book is energy-efficient NN processor integrating digital circuits and CIM IPs. Aiming at the shortcomings of existing NN processors and the unexplored design space, systematic research work is carried out. This book contains four main parts of research contents (Chaps. 3–6), as shown in Fig. 1.3. The first part mainly studies the data reuse optimization, which designs a specific architecture optimization for the specific convolution kernel to achieve more efficient data reuse. In the second part, a regular NN model compression method is adopted, and the frequency-domain compression algorithm is used to achieve higher energy efficiency. In the third part, based on the research on digital circuit NN processor in the first two parts, the data reuse and model compression technologies are transferred to the CIM chip design. Data reuse and sparsity techniques in CIM chips are explored to combine the benefits of digital circuits and CIM IPs. The fourth part research combines the requirements of the actual NN application, further optimizes the sparse utilization, and considers the weight update cost, which further improves the energy efficiency of the actual system. It can be expanded to support larger-scale NN models.

Focusing on the theme of "High energy efficiency neural network processor with combined digital and computing-in-memory architecture", the four parts of the research work gradually go deeper with an obvious progressive relationship. The first part starts from the data reuse optimization, which deeply studies the data reuse opportunities in the NN algorithm. The relevant data reuse technologies are adopted in all the second, third and fourth parts. The second part explores the compression optimization of the NN model, and adopts a structured frequency-domain compression algorithm. Based on the first part, an efficient two-dimensional data reuse is

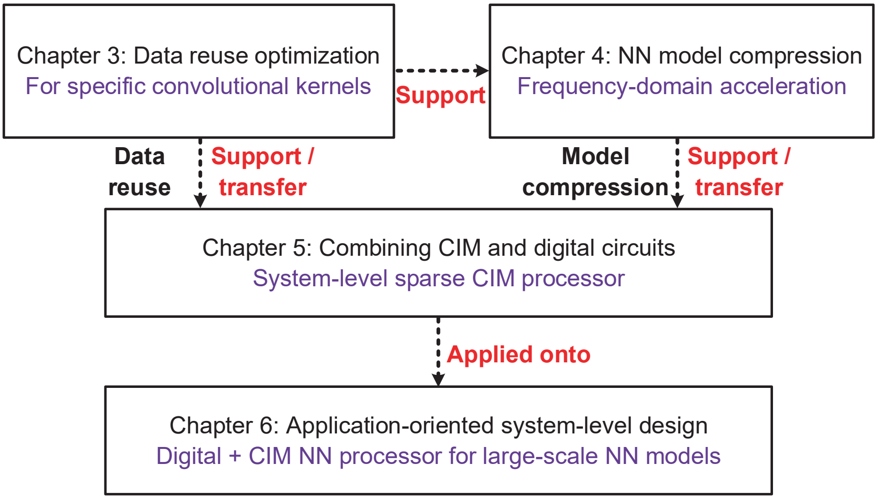
Chapter 3: Data reuse optimization
For specific convolutional kernels
Support
Chapter 4: NN model compression
Frequency-domain acceleration
Data reuse
Support / transfer
Model compression
Support / transfer
Chapter 5: Combining CIM and digital circuits
System-level sparse CIM processor
Applied onto
Chapter 6: Application-oriented system-level design
Digital + CIM NN processor for large-scale NN models

**Fig. 1.3** Overall framework of the research work

realized in the frequency-domain computing array. In order to achieve higher energy efficiency, the third part converts from pure digital circuits to the system-level chip design with combined digital and CIM circuits, which inherit the technical ideas of data reuse and model compression in the first two parts. It explores the data reuse technology in the CIM system, and proposes the structured block-wise sparsity strategy on the CIM architecture. The fourth part further extends the third part, and proposes the set-associate block-wise sparsity. Meanwhile, for the weight updating problem when implementing large-scale NN models on small CIM chips, a ping-pong mechanism to support simultaneous weight updating and CIM operations is proposed, so that the CIM chip can be extended to support large-scale NN models. Overall, the four parts of the research work are deepening around the same theme, with strong connections and progressive relationships. The technical solutions of each part are combined to realize an energy-efficient NN processor with combined digital and CIM circuits.

### *1.2.2 Main Contributions of This Book*

The main contributions and achievements of this book are as follows.

- **Propose a data reuse optimization architecture based on specific convolution kernel**. By trade-off with the flexibility of different convolution kernel sizes, a computing array dedicated to a specific convolution kernel is designed to achieve more efficient data reuse. Compared with two representative NN chips, it reduces 4.9× SRAM access overhead or 16.7× SRAM area, and achieves 4× energy efficiency.

- **Implement a NN processor with regular frequency-domain compression and acceleration**. In order to solve the significant additional power/area overhead incurred by irregular sparsity, a high-performance low-power fast Fourier transform (FFT) circuit and a frequency-domain two-dimensional data-reuse computation array based on transpose SRAM (TRAM) are designed by adopting a structured frequency-domain compression algorithm. The test chip was fabricated in TSMC 65 nm process. Compared with the representative NN chip, the area efficiency and energy efficiency are 8.1× and 4.2×, respectively.
- **Explore system-level CIM chip design with sparsity technique**. A CIM chip supporting block-wise sparsity is designed, and an inter/intra-macro data reuse technology for CIM architecture is proposed. This work has fabricated a CIM chip in TSMC 65 nm process, which achieves 35TOPS/W peak energy efficiency.
- **Explore data update operations and large network expansion of the CIM architecture**. The set-associate block-wise sparsity technology, ping-pong CIM and weight update technology are proposed. The adaptive precision analog-to-digital converter (ADC) is also explored. This work fabricated a CIM chip in TSMC 65 nm process, which achieves 6.3× energy efficiency compared with the state-of-the-art CIM chip.

## 1.3 Overall Structure of This Book

This book investigates the wide application requirements of NN algorithms on low-power intelligent devices, and explores energy-efficient NN processors that integrate digital and CIM circuits. The organization of this book is as follows.

This chapter introduces the background of artificial intelligence and neural network. It points out the significance of energy-efficient NN processor for a wide range of low-power intelligent devices, and briefly introduces the current research status and shortcomings of NN processors based on digital or CIM circuits, which then motivates the main contents and contributions of this book.

Chapter 2 introduces the basics and related research work of NN algorithms and NN processors. Firstly, the basic structure of NN algorithms and the challenges of energy-efficient NN chip are introduced. Then, the related optimization technologies of NN processors based on digital circuits and CIM architectures are introduced, respectively. The advantages and disadvantages are analyzed. Besides, the potential optimization space of integrating digital and CIM circuits are pointed out.

Chapter 3 introduces the data reuse optimization architecture for specific convolution kernel. This chapter first analyzes the insufficient data reuse caused by the traditional NN processors' equal support for different convolution kernels, and then introduces the three main techniques: Make data reuse more sufficiently by (1) optimizing the processing array for a specific convolution kernel and (2) the cyclic access of the local memory; (3) Realize module-level parallelism through the instruction set of one-instruction-multi-cycle execution, which improves high utilization rate and high average energy efficiency. Finally, this chapter shows the chip layout based on the above techniques and the improvement compared with the representative work, and analyzes the advantages and disadvantages of this design.

Chapter 4 introduces the NN chip based on a frequency-domain structured compression algorithm. This chapter first analyzes the significant additional power/area overhead introduced by irregular sparse compression, and then introduces the frequency-domain structured compression acceleration algorithm. In this chapter, a high-performance low-power global-parallel bit-serial FFT circuit, a block-wise TRAM and a frequency-domain two-dimensional (2D) data reuse processing array are designed to realize an energy-efficient frequency-domain NN chip. Finally, this chapter shows the improvement compared with the representative work, and explains the advantages and disadvantages.

Chapter 5 introduces a sparse NN chip that combines digital and CIM circuits. This chapter first analyzes the design deficiencies of system-level CIM chips, and introduces the challenges of implementing sparsity and data reuse techniques in a CIM system. Then, this chapter introduces the block-wise weight sparsity and dynamic activation sparsity, flexible network mapping and intra/inter-macro data reuse techniques. Besides, the CIM macro (IP) is designed to support dynamic power-off ADC. Finally, this chapter shows the test results of the fabricated CIM chip and the comparison with the representative work.

Chapter 6 introduces the application-oriented “Digital + CIM” NN chip. This chapter first analyzes the challenges of insufficient sparsity utilization and difficulty in supporting large-scale NN applications in the CIM architecture, and then introduces the set-associate block-wise sparsity, ping-pong CIM and weight update, and the CIM macro with adaptive precision. Finally, this chapter shows the test results and comparison, and analyzes the improvement on the large-scale NN model, showing the advantages and future improvement of this design.

Chapter 7 summarizes the research and main contributions of this book, and analyzes the development trend and application prospect of energy-efficient NN processors.

## References

1. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444
2. The State Council of the People’s Republic of China (2017) Development plan for the new generation of artificial intelligence, no 35
3. Dutton T (2018) An overview of national AI strategies
4. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) ImageNet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp 248–255
5. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
6. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556

7. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2014) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
8. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
9. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
10. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Proceedings of the 31st international conference on neural information processing systems, pp 6000–6010
11. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, 27
12. Silver D, Huang A, Maddison CJ, Guez A, Sifre L, Van Den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M et al (2016) Mastering the game of go with deep neural networks and tree search. Nature 529(7587):484–489
13. Ltd. Askci Consulting Co. (2020) 2020–2025 China's artificial intelligence chip industry outlook forecast and market research report
14. Bianco S, Cadene R, Celona L, Napoletano P (2018) Benchmark analysis of representative deep neural network architectures. IEEE Access 6:64270–64277
15. Niu W, Ma X, Lin S, Wang S, Qian X, Lin X, Wang Y, Ren B (2020) PATDNN: achieving real-time DNN execution on mobile devices with pattern-based weight pruning. In: Proceedings of the 25th international conference on architectural support for programming languages and operating systems, pp 907–922
16. NVIDIA A100 tensor core GPU (2021)
17. JETSON NANO (2021)
18. The 19th Central Committee of the Communist Party of China (2020) Proposal of the central committee of the communist party of China on formulating the 14th five year plan for national economic and social development and the long range goals for 2035. People's Publishing House
19. Chen T, Du Z, Sun N, Wang J, Wu C, Chen Y, Temam O (2014) DIANNAO: a small-footprint high-throughput accelerator for ubiquitous machine-learning. In: Proceedings of the 19th international conference on architectural support for programming languages and operating systems, pp 269–284
20. Chen Y-H, Krishna T, Emer J, Sze V (2016) 14.5 Eyeriss: an energy-efficient reconfigurable accelerator for deep convolutional neural networks. In: 2016 IEEE international solid-state circuits conference (ISSCC). IEEE, pp 262–264
21. Bang S, Wang J, Li Z, Cao G, Sylvester D (2017) 14.7 A 288 $\mu$w programmable deep-learning processor with 270 kB on-chip weight storage using non-uniform memory hierarchy for mobile intelligence. In: 2017 IEEE international solid-state circuits conference (ISSCC). IEEE, pp 250, 251
22. Zhou S, Wu Y, Ni Z, Zhou X, Wen H, Zou Y (2016) Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. arXiv:1606.06160
23. Rastegari M, Ordonez V, Redmon J, Farhadi A (2016) Xnor-net: ImageNet classification using binary convolutional neural networks. In: European conference on computer vision. Springer, pp 525–542
24. Han S, Pool J, Tran J, Dally W (2015) Learning both weights and connections for efficient neural network. In: Advances in neural information processing systems, pp 1135–1143
25. Wei W, Wu C, Yandan W, Yiran C, Hai L (2016) Learning structured sparsity in deep neural networks. Adv Neural Inf Process Syst 29:2074–2082
26. Mao H, Han S, Pool J, Li W, Liu X, Wang Y, Dally WJ (2017) Exploring the granularity of sparsity in convolutional neural networks, pp 13–20
27. Zhang T, Ye S, Zhang K, Tang J, Wen W, Fardad M, Wang Y (2018) A systematic DNN weight pruning framework using alternating direction method of multipliers. In: Proceedings of the European conference on computer vision (ECCV), pp 184–199