International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

Low Power FPGA Architecture for Deep CNN to Predict Lung Tumor

Nistula Jayasri P1, Nivetha K1, Sagaya Agnes Rayan T1 and Mr. Balasundaram S2 M.E.

1Students, B.E. Electronics and Communication Engineering
2Assistant Professor, Department of Electronics and Communication Engineering
3Meenakshi Sundararajan Engineering College, Kodambakkam, Chennai, Tamil Nadu, India.

Abstract - *Most of the medical diagnosis depends on the CT and MRI scanned images and their predictions. In many test cases, the prediction of tumors and other critical diseases depends on the high-resolution imaged predictions and predicting systems. In the existing system, various types of disease identification, prediction through CT, MRI images are discussed using MATLAB-based image processing support. The tool is meant best for image processing applications, henceforth the design model provides efficient results on enhancing the same. In the proposed system, a low-power VLSI architecture is evaluated in which the FPGA platform acts as reconfigurable hardware which holds the detailing features of the preprocessed input test image and predicts the disease, to classify to some extent accurately. The benefits of FPGA on operating speed, the reconfigurable structure, and the low power platform extend the motivation of work to develop it in the XILINX platform. Simulations and synthesis are done in ModelSim 6.3 version and coding is done in Verilog HDL.*

Key Words: **FPGAs, hardware accelerator, CNN, VGM, MAC, lung tumor.**

1. INTRODUCTION

Lung cancer is a disease wherein abnormal cells multiply and grow into a tumor. Cancer cells are carried away by the blood, or lymph fluid that surrounds the lung tissue. Lymph flows through lymphatic vessels via ducts into lymph nodes. Lymph nodes are located in the lungs and are in the center of the chest. As lymph flows out of the lungs, towards the center of the chest, lung cancer often spreads towards the center of the chest. As a cancer cell leaves the site where it occurred and moves into a lymph node or any other part of the body through the bloodstream as a medium, metastasis ensues [1]. Cancer that emerges in the lungs is called primary lung cancer. There are several different types of lung cancer categorized into two main groups namely, small cell lung cancer and non-small cell lung cancer which are subdivided into three types: Carcinoma, Adenocarcinoma, and Squamous cell carcinomas. A survey conducted in 2008 among Jordanians indicated that, in the rank of order for both males and females, there were 356 cases of lung cancer accounting for 7.7% of all newly diagnosed cancer cases. The male-to-female ratio of 5:1 was recorded as 297 (13.1%) males and 59 (2.5%) females were affected. Lung cancer ranked second among males and tenth among females [2]. Based on the stage of discovery of cancer cells in the lungs, lung cancer is considered the most dangerous and widespread cancer in the world. Hence the process of early detection plays a vital role to reduce its percentage of distribution to avoid seriously advanced stages. The primary objective of our project is to design and implement an efficient VLSI low power architecture in an FPGA to predict Lung Tumor.

2. GRAPHICS PROCESSING UNIT (GPU)

The graphics processing unit is becoming one of the vital bases of modern supercomputing. They are capable of rapid processing of data mainly for interpretation of images. Their usage in upcoming hyper-scale data centers has led them to be accelerators, which is capable to speed up all sorts of tasks such as networking, encryption, artificial intelligence, etc., [3]. With the help of these GPUs, there have been many advances in gaming as well as in the graphic industry. Algorithms that require processing immense blocks of data in parallel need a processor which is capable of parallel structure. This is where GPU is more beneficial than Central processing units (CPUs) [4]. To facilitate processing multiple videos at one on supercomputers, workstations, and 3D rendering, in the case of VFX and for simulations and training workloads using AI multiple GPUs are utilized. For example, NVIDIA GPUs, in contrast to CPUs, contain chips implanted with CUDA cores, and each of these is a tiny processor that executes some code.

Fig- 1: NVIDIA chip.

2.1 LIMITATIONS

- The incorporation of GPU hardware into systems increases the expense in terms of power consumption, heat production, and cost.

- To obtain speed, the algorithm coded must be suitable for GPU architecture, but the programming of GPU differs entirely from the programming of CPU.

- The incorporation of GPU acceleration to pre-existing codes is more strenuous than moving from one CPU family to another.

3. VISION PROCESSING UNIT (VPU)

A vision processing unit (VPU) is a type of microcontroller and a specific kind of AI accelerator that is meant for machine vision tasks. These are more suitable for performing machine vision algorithms [5]. They are fabricated for parallel processing and are designed with resources to capture and obtain visual data from the cameras. These can be connected to other interfaces for programmable use as they consume low power and provide high performance. They are designed for the acceleration of machine vision algorithms such as CNN (convolutional neural networks), SIFT (Scale-invariant feature transform), and similar ones. They possess direct interfaces to obtain data from cameras and emphasize on-chip data flow between many parallel execution units. Growth in the usage of smartphones, increasing adoption of edge AI, and the soaring demand for advanced capacities for computer vision, the internet of things, robotics are the driving factors for VPU development [6]. One example of a VPU is Intel's Movidius Myriad X VPU which is being used in many of the edge devices.

Fig- 2: Movidius Myriad X VPU

3.1 LIMITATIONS

- Comparatively less powerful than GPU due to its compact size for mobile applications.

- As it is not widely used currently, performance and various features are unknown.

4. APPLICATION-SPECIFIC INTEGRATED CIRCUIT (ASIC)

With the advent of Application-Specific Integrated Circuits, a whole class of AI hardware accelerators is gaining eminence. Application-Specific Integrated Circuits are integrated circuits designed to perform a specific task. ASICs are the emerging technology to deliver AI compute. Optimum memory usage and utilization of lower precision arithmetic are the strategies employed to accelerate calculation and increase the throughput of computation. Some adopted low-precision floating-point formats used AI accelerations are half-precision and the bfloat16 floating-point format. Hardware acceleration is used to expedite the computing processes present in an AI workflow. One of the prime advantages of ASIC is speed, as accelerators minimize the time taken for training and execute AI model or AI-based tasks [7]. For example, Intel released Nervana, an ASIC for inference and support for a large amount of parallelization in server settings. It has also restructured the chip considerably and built them on a 10nm manufacturing process.

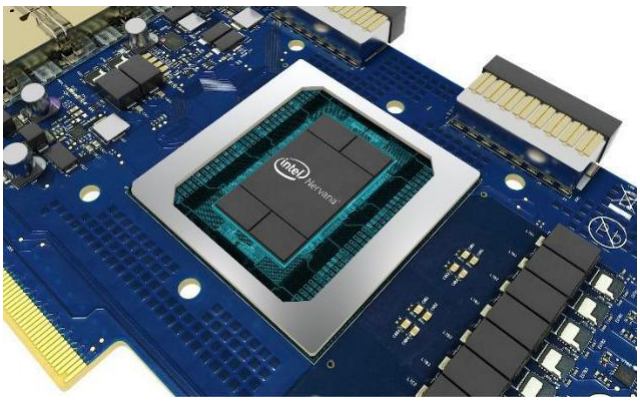

Fig- 3: Nervana ASIC

4.1 LIMITATIONS

- As ASIC is a dedicated hardware performing a single task, the addition of new features or patching bugs can be tedious or maybe impossible leading to the replacement of hardware.

- It is harder and costlier to develop new hardware than software.

5. TENSOR PROCESSING UNIT (TPU)

A tensor processing unit is a circuit specially designed to implement all the fundamental controls and arithmetic logic required to implement machine learning algorithms, often operating on predictive models such as Artificial Neural Networks (ANNs) or Random Forests (RFs). Tensors are matrices or multi-dimensional arrays and are the elemental units that hold data points similar to node weights in a

neural network in a row and column format [8]. Tensors perform basic calculation operations. TPUs were employed in the known DeepMind's AlphaGo, where the world's best Go player was conquered by AI. It was also utilized in the AlphaZero system, which produced programs such as Chess, Shogi, and Go-playing. TPUs was launched by Google in the year 2016. TPUs, unlike GPUs, are custom-designed to utilize operations like matrix multiplications in neural network training. The power of Google TPUs can be reached in two types, which are cloud TPU and edge TPU. Cloud TPUs may be accessed from Google Colab notebook, which gives users TPU pods that sit on Google's data centers. On the other hand, edge TPU is a custom-built development kit that can be utilized to create specific applications.

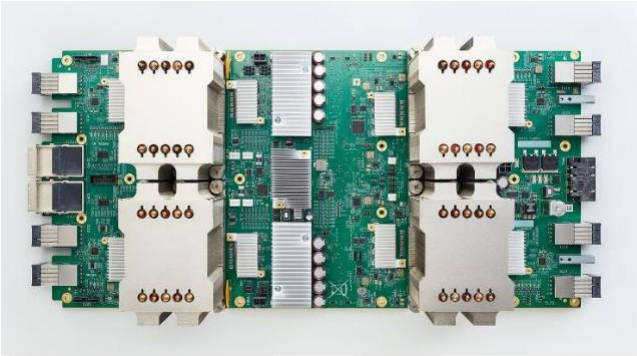

Fig- 4: Cloud TPU

Fig- 5: Edge TPU

5.1 LIMITATIONS

- The topology of TPU is different from other hardware platforms and it is not easy to work with for those who aren't familiar with DevOps and the peculiarity of the TPU.
- It only supports TensorFlow currently and operations such as customer operations written in C++ are not supported.

6. FIELD-PROGRAMMABLE GATE ARRAY (FPGA)

Field-Programmable Gate Array are semiconductor devices. These are integrated circuits that can be reprogrammed to desired application or functionality requirements after manufacturing hence the name "field-programmable". FPGA is based around a matrix of configurable logic blocks connected via Programmable interconnects that enable the blocks to be connected which resembles logic gates that can be inter-wired in various configurations. It has nearly the same efficiency as application-specific integrated circuits (ASICs) and is as flexible as a CPU. It is favorable over GPU in the case of interface flexibility and is augmented by the integration of programmable logic with CPUs and standard peripherals. Current FPGAs have large resources of logic gates and RAM blocks to realize complex data computations. Due to their reconfigurable characteristics, FPGAs are considered a perfect fit in various markets. This specific feature contrasts FPGAs from Application-Specific Integrated Circuits (ASICs), which are for particular tasks and are custom produced [9]. FPGAs are applied to accelerate AI workloads in data centers for machine learning inference jobs. Numerous hardware companies such as Xilinx have launched their FPGA products as the latest datacenter accelerator cards as satisfying increasing business demand for heterogeneous architectures and performance advances as customers work on more AI workloads.

Fig- 6: Xilinx FPGA Module

7. PROPOSED FRAMEWORK

In the existing system, sparse-wise dataflow is used. Here the cycles of processing multiply-and-accumulates (MACs) are skipped with zero weights and to minimize energy it exploits data statistics through zero gating to avert unnecessary computations. The sparse-wise dataflow employed leads to a low bandwidth requirement and high data sharing. Then an FPGA accelerator is designed comprising of a vector generator module (VGM) that matches the index between sparse weights and input activations [10]. The convolution

neural network (CNN) architecture is implemented using VHDL and as an application model, identification of lung tumors is considered. The proposed FPGA architecture predicts lung cancer through the low-power designed CNN architecture. Modelsim 6.3 is used for the simulation of the architecture. Current identification methods to identify lung tumors include MATLAB-based image processing techniques. We aim at constructing a low-power architecture by using low-power techniques such as clock gating and power gating as it has not been attempted. Low-power VLSI technique is used to detect the area where the lung tumor is present in the image given as an input. Initially, the datasets were collected and MATLAB software is used to extract the features of each image and convert it as a binary value so that it can be trained further using VHDL programming in ModelSim to predict the area where lung tumor is present. In the pre-processing stage, the input image undergoes regional segmentation, where we group pixels, or sub-regions into larger regions. Then we set the start of the seed point, after which it automatically groups the surrounding pixels, which have similar properties to that of the seed point. The segmented image is converted to grayscale for further processing of the Blob Image where the required pixels are found and labeled. Both the processed images are fused and again converted to grayscale. The extracted data, in the form of binary, is used in the VHDL to train and predict the area of the tumor. A digital convolution architecture is designed with the help of clock-gated latches, which perform the CNN on the data. The metric scores are generated and compared to predict the area of the tumor. Based on this, a low-power VLSI model to predict the presence of lung tumors in an image has been developed.

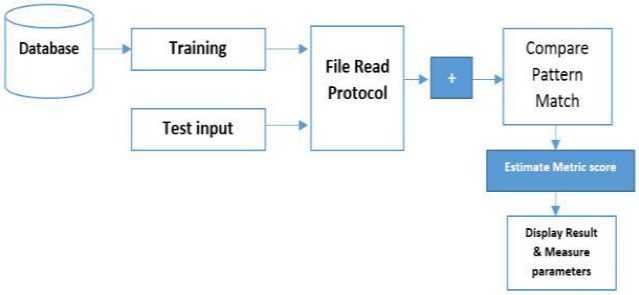

Fig- 7: System Architecture

8. RESULTS AND DISCUSSIONS

The following are the images obtained after the preprocessing stage in MATLAB. Each image represents the consecutive stages involved in the preprocessing of the input image.

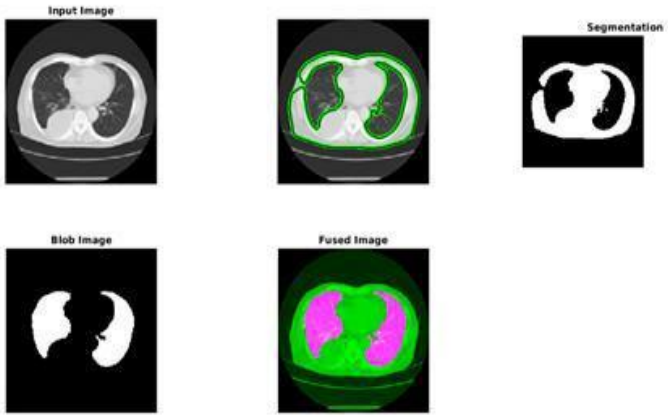

Fig- 8: Output from preprocessing stage

The output obtained from ModelSim 6.3 is given below. The predicted disease is highlighted in the left pane.

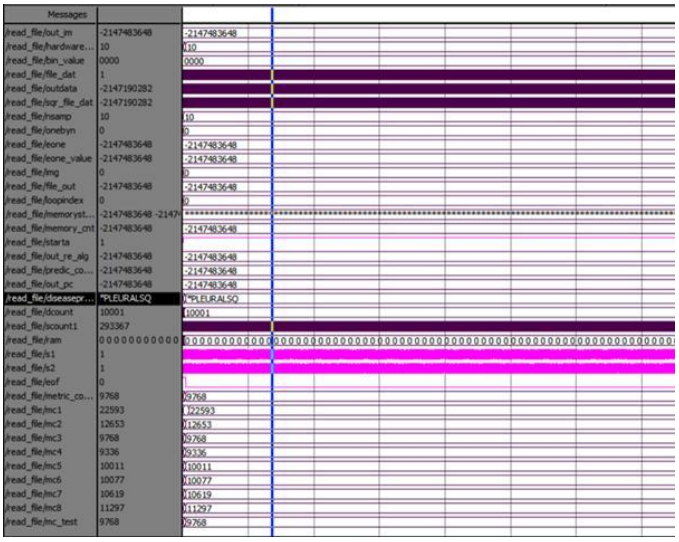

Fig- 9: Simulation Output

The data flow diagrams of the architecture is given as follows. The output of the hardware_control of the disease prediction block can be connected externally to the hardware kit to display the results.

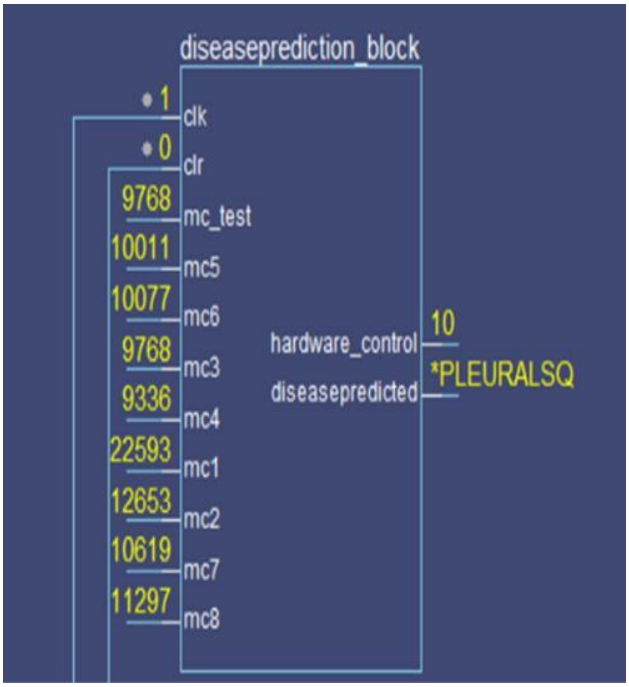

Fig- 10: Prediction block

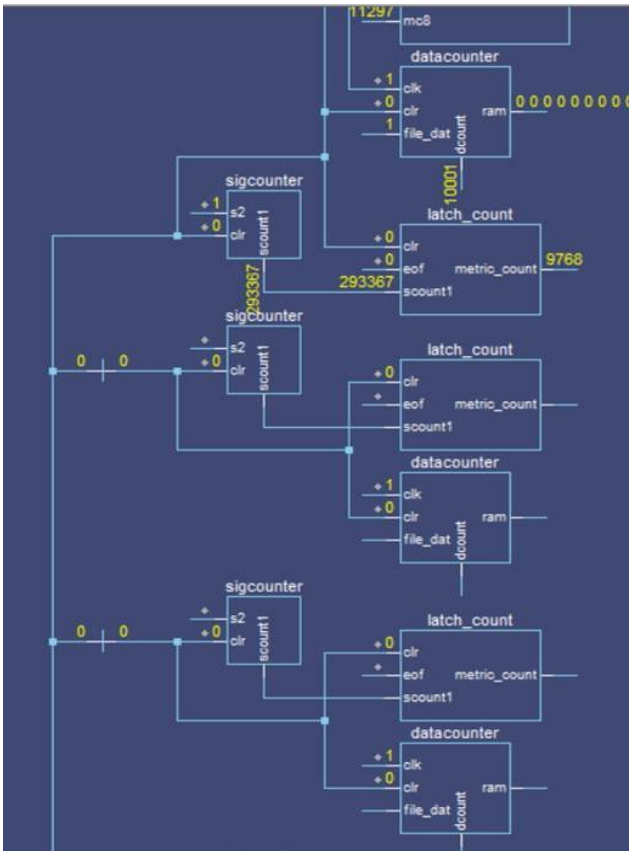

Fig- 11: Metricscore generation block

9. FUTURE SCOPE

In the coming future, we deem this application that diagnoses lung diseases to determine technology in the healthcare field and it can promote detecting lung cancer with more accuracy. In the medical field, there is more chance to develop or convert this project in numerous approaches. Thus, this project has an efficient scope in the coming future, where manual predicting can be cheaply and efficiently converted to computerized production.

10. CONCLUSION

This project is developed to detect the presence of lung diseases using the VLSI technology. This also helps in providing efficient treatment in the most economical means and eventually reduces the time required for identifying lung diseases in comparison to the current technique, as FPGAs are considerably more efficient than CPUs. Diagnosing manually consumes more time and also involves human error rate. Thus, this project eliminates the human error rate and reduces the time required for manual classification with a low power feature.

REFERENCES

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.

[2] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1026–1034.

[3] Liu B, Qiu W, Jiang L, Gong Z. Software pipelining for graphic processing unit acceleration: Partition, scheduling and granularity. The International Journal of High Performance Computing Applications. 2016;30(2):169-185.

[4] S. Asano, T. Maruyama, Y. Yamaguchi; "Performance comparison of fpga, gpu and cpu in image processing," in International Conference on Field Programmable Logic and Applications(FPL), 2009.

[5] Z. Du et al., "An accelerator for high efficient vision processing", IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 36, no. 2, Feb. 2017.

[6] B. Barry, C. Brick, F. Connor, D. Donohoe, D. Moloney, R. Richmond, et al., "Always-on vision processing unit for mobile applications", IEEE Micro, vol. 35, no. 2, 2015.

[7] E. Nurvitadhi et al., "Accelerating recurrent neural networks in analytics servers: Comparison of FPGA CPU GPU and ASIC", FPL, 2016.

[8] Jouppi, N. P. et al. In-datacenter performance analysis of a tensor processing unit.

[9] I. Kuon and J. Rose, "Measuring the Gap Between FPGAs and ASICs," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 26, no. 2, Feb. 2007.

[10] Chaoyang Zhu, Kejie Huang, Shuyuan Yang, Ziqi Zhu, Hejia Zhang and Haibin Shen, "An Efficient Hardware Accelerator for Structured Sparse Convolutional Neural Networks on FPGAs", Jan. 2020.