



Area efficient low power VLSI of 2048-Point pipelined radix 16 MDC /FFT Processor for brain tumour detection using optimized deep dilated convolutional neural network

L.Mohana kannan^{a,*}, Rama Chaithanya Tanguturi^b, Parul Dubey^c, D. Haripriya^d

^a Department of Biomedical Engineering, Erode Sengunthar Engineering College, Thudupathi, Erode-638057, Tamil Nadu, India

^b AI Research Centre, Woxsen University, Hyderabad, Telangana, India

^c Department of Computer Science and Engineering, Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Pune, India

^d Department of Computer Science and Engineering, Veltech Rangarajan Dr.Saguntala R&D Institute of Science and Technology, Avadi, Chennai, Tamil Nadu, India

ARTICLE INFO

Keywords:

Fast Fourier Transform
Very Large Scale Integration
Brain Tumor Classification
Deep Learning
Mixed-Difference Component

ABSTRACT

This research presents a novel, area-efficient, low-power VLSI design for a 2048-point pipelined Radix-16 Mixed-Difference Component (MDC) FFT processor, specifically aimed at enhancing brain tumor detection with MRI. Addressing common FFT design challenges such as power consumption, computational delay, and area efficiency, the design integrates Approximate Carry Disregard Multipliers (ACDM) and a Low-Power Pass-Transistor Logic-Based Full Adder (LPTLFA). Additionally, the system employs a Deep Dilated Convolutional Neural Network (DDCNN) optimized with the Fire Hawk Optimizer (FHO) to improve classification accuracy by fine-tuning the loss function. This advanced architecture significantly reduces power usage, silicon area, and computational delay, making it suitable for resource-constrained environments. Experimental results demonstrate that the proposed Fast Fourier Transform (FFT) processor achieves a worst-case delay of 11.43 ps and outperforms existing solutions in speed and power efficiency, offering a promising solution for advanced medical imaging systems.

1. Introduction

Brain tumors are among the most life-threatening forms of cancer, and their early detection is crucial for improving patient outcomes. Accurate diagnosis typically relies on advanced imaging techniques such as MRI and CT scans, which generate large volumes of data that must be processed efficiently. Signal processing plays a critical role in this context, with the Fast Fourier Transform (FFT) being one of the most essential algorithms used for analyzing these complex datasets [1,2]. The FFT is fundamental in converting time-domain signals into their frequency-domain counterparts, making it invaluable in medical imaging applications. However, implementing FFTs in real-time medical diagnostics, particularly for brain tumor detection, presents several significant challenges, particularly in designing Very Large Scale Integration (VLSI) systems that are both power-efficient and area-efficient. The need for area-efficient and low-power VLSI designs has become increasingly important as medical devices continue to evolve towards

portability and real-time processing capabilities. These devices must be capable of performing high-speed computations while adhering to strict constraints on power consumption and physical chip area. The FFT processor, as a core component in these systems, must therefore be optimized to meet these demands without compromising performance. One of the primary challenges in this domain is implementing large-point FFTs, such as the 2048-point FFT, which is required for high-resolution imaging used in brain tumor detection. The complexity of such an FFT increases significantly with the number of points, making it challenging to design an efficient VLSI implementation [3–6].

Medical imaging, especially MRI, is one of the significant backbones of modern healthcare as it may help in the early detection of critical conditions, such as those resulting from brain tumors. Advanced medical imaging systems are in high demand, especially in resource-constrained settings like rural healthcare facilities portable diagnostic systems, and real-time monitoring devices. However, the existing approaches still face significant challenges in terms of computational efficiency, power

* Corresponding author.

E-mail addresses: mohancalls2000@gmail.com (L.Mohana kannan), trchaitanya@gmail.com (R.C. Tanguturi), dubeyparul29@gmail.com (P. Dubey), drharipriya@veltech.edu.in (D. Haripriya).

<https://doi.org/10.1016/j.measurement.2025.116691>

Received 14 September 2024; Received in revised form 11 December 2024; Accepted 6 January 2025

Available online 9 January 2025

0263-2241/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

consumption, and hardware scalability. Generally, the various limitations in these aspects often tend to restrict the implantation of high-performance imaging systems when energy, space, and computational resources are restricted. Signal processing is the back-bone of MRI image analysis while FFT processors constitute an important part. However, traditional FFT architectures are quite hardware-intensive, characterized by high power consumption, large silicon area, and significant computational delays that limit their suitability to portable or embedded systems. Simultaneously, the integration of deep learning models into medical imaging workflows adds complexity; as such models may require abundant computational resources and optimal fine-tuning toward reliable and interpretable outcomes. The challenges of designing an efficient FFT processor for brain tumor detection are multifaceted. Firstly, achieving a balance between speed, power consumption, and silicon area is difficult. High-radix FFT algorithms, such as Radix-16, are commonly used to reduce the number of computational steps required. While these algorithms can significantly enhance processing speed, they also increase the complexity of the hardware, leading to greater power consumption and a larger chip area. These factors are particularly problematic in the context of portable medical devices, where power efficiency and minimal size are paramount. Various approaches have been developed to address these challenges, with techniques such as pipelining and parallel processing being among the most prominent. Pipelining, for instance, allows for the overlapping of computation stages, thereby increasing throughput. However, traditional pipelined architectures, particularly those based on the Multi-path Delay Commutator (MDC) approach, face limitations in optimizing both power efficiency and area. While MDC-based FFT processors are known for their high throughput, they often require significant silicon area and consume considerable power, which limits their suitability for resource-constrained environments like portable medical devices. Another challenge lies in the integration of FFT processors within the broader system architecture of medical devices, particularly in conjunction with other signal-processing components. Ensuring that the FFT processor operates efficiently within the overall system, without becoming a bottleneck, is crucial. However, existing approaches often fall short of achieving this integration seamlessly, leading to inefficiencies and potential delays in real-time applications [7–11].

The drawbacks of existing methods highlight the need for further research in developing FFT processors that are not only fast and accurate but also optimized for low power consumption and minimal area usage. The increasing complexity of medical imaging systems, combined with the need for portable and real-time diagnostic tools, underscores the importance of this research area. Developing innovative solutions that address these challenges is essential for advancing the field of medical diagnostics, particularly in the early detection and treatment of brain tumors. While significant progress has been made in the design of FFT processors for medical applications, several challenges remain, particularly in achieving the required balance between speed, power efficiency, and silicon area [12–15]. The ongoing need for improvements in this field drives the pursuit of new research directions that can overcome the limitations of existing approaches, ultimately contributing to more effective and accessible brain tumor detection technologies. Traditional methods in the detection of a brain tumor rely on computationally intensive processes, which require high-power consumption, large silicon areas, and significant delays in computation. Those constraints along with dimensionality make it challenging to integrate such systems into resource-constrained environments, such as portable medical devices or real-time systems. With the design of the 2048-point pipelined Radix-16 MDC FFT processor using ACDM and LPTLFA, this work presents considerable reductions in power, area, and delay. The inclusion of a DDCNN in the design helps it accurately detect tumors by capturing multi-scale patterns from the frequency domain data. The FHO optimizes the model, giving a high classification performance by efficiently tuning the loss function. This leads to an improvement in detection capability while maintaining the low power and area requirements, thus

making it more practical for use in real-world medical applications. The study not only highlights the relevance of bridging the identified gaps but also focuses on a broader impact on healthcare, particularly in improving access to reliable diagnostics in underserved regions. This work represents a significant step forward towards integrated hardware-software solutions that bear relevance to real-time, efficient, and accurate medical imaging applications.

2. Novelty and contributions

The novelty of the proposed FFT architectures comes in the fact that the design embeds state-of-the-art hardware optimizations from the domain's highest profile deep learning techniques, particularly oriented toward medical imaging applications. This differs substantially from most traditional architectures of high-performance FFT designs, which either focus on speed or are power-efficient. Instead, this design provides all three: very low power consumption, reduced delay in computation, and a minimal silicon area by deploying Approximate Carry Disregard Multipliers and Low-Power Pass-Transistor Logic-Based Full Adders. The deep dilated convolutional neural network optimized by Fire Hawk Optimizer would be the synergistic interface between FFT processing and machine learning towards achieving high accuracy in tumor classification using better computational efficiency. Novel hardware juxtaposed with AI-driven optimization makes it a pioneering resource-constrained medical system in itself, achieving high accuracy and forming a new benchmark for FFT architectures. The major highlights are provided below.

- The use of Approximate Carry Disregard Multipliers (ACDMs) in the FFT processor significantly reduces power consumption and area compared to traditional multipliers. Unlike conventional multipliers that require complex and power-hungry circuitry to handle carry propagation, ACDMs strategically ignore certain carry bits, simplifying the hardware and speeding up the computation. This choice is particularly advantageous in the design of a high-performance VLSI system, where minimizing power consumption and silicon area is critical.
- The PTL-based Full Adder was selected for its superior power efficiency and reduced transistor count compared to conventional CMOS full adders. The PTL approach minimizes the number of logic gates required, thereby decreasing both dynamic and static power consumption.
- The DDCNN was chosen for its ability to capture multi-scale contextual information in MRI images, which is essential for accurate brain tumor detection. Unlike standard convolutional networks, the DDCNN employs dilated convolutions that allow for an exponentially expanding receptive field without increasing the number of parameters or computational load. Furthermore, the DDCNN was optimized using the FHO, enhancing the network's performance by fine-tuning the loss function for better convergence and accuracy.

The organization of this research is as follows: [Section 2](#) provides a comprehensive summary of prior studies concerning obstacle detection for individuals with visual impairments. In [Section 3](#), the proposed model is elaborated in detail. [Section 4](#) examines the outcomes and discussions. [Section 5](#) summarizes the research and outlines future scopes.

3. Review

Numerous studies have explored the design of FFT processors for various applications. The following review provides an overview of some of these studies, highlighting their methodologies and contributions to the field.

An efficient hardware implementation approach for a variable-size fast Fourier transform (FFT) processor for spectral analysis was

suggested by Hazarika J et al. [7]. The processor was designed to handle different frame sizes, making it adaptable to various standard requirements. A serial real-valued processor with a new data-flow graph was considered, as it minimized the number of required multipliers. Stage-specific optimization techniques and a multiplierless structure were jointly employed to enhance hardware efficiency. Clock gating was utilized to enable variable-size operation and reduce power consumption. A Fixed-Point (FP) analysis of the design was conducted. The novel architecture relied on a Shift and Accumulation (SA) technique, with partial products generated and shared based on their symmetries. A hybrid radix encoder was developed by Priyadharsini R and Sasipriya S [16], incorporating a Radix Encoder with a Two-stage Operand Trimming Logarithmic appropriate multiplier and an Optimized Truncated Kogge-stone adder (RETOTL-OTK) for a 2048-point and 4096-point FFT processor intended for FPGA implementation. The design focused on achieving high throughput with minimal power consumption. The processor was engineered to handle FFT sizes from 16 to 4096 points, targeting both biomedical applications and emerging 5G technology. The significant demands of high-throughput medical image processing were highlighted by Vinayagam P et al. [17]. Challenges inherent to computational aspects of medical imaging were discussed, underscoring the necessity of overcoming these obstacles. The architecture, parallelization strategies, and processing pipeline were described, providing insight into the underlying hardware and software. Evaluation results demonstrated substantial improvements in processing speed and image quality, particularly in real-time applications. Deep learning (DL) techniques were combined by Malathi L et al. [18] with FFT to address image quality and resolution challenges. A Convolutional Neural Network (CNN) was utilized to enhance image quality, particularly by recovering lost information. The process involved de-noising through the Double Density Discrete Wavelet Transform (DDDWT) technique, followed by segmentation and culminated in resolution enhancement using a CNN integrated with an Overlap-and-Add (OA)-FFT-based Wallace Tree multiplier and Red Deer Optimization (WT-RDO). An end-to-end edge-enabled VLSI architecture based on machine learning was introduced by Parmar R et al. [19] for classifying ECG excerpts with Atrial Fibrillation (AF) from normal beats. It was noted that abnormal atrial activity has been historically observed within the low-frequency range, a focus not previously explored in this context. The proposed work analyzed this frequency band directly for AF detection.

Sadaghiani AK and Forouzandeh B [20] introduced a high-performance power spectrum and bispectral density analyzer for noisy biomedical signals. A Radix-8 Memory-based 1024-point Blackman-Tuckey (R8M-BT) method was used for power spectral density estimation. The design featured a shared-resource CORDIC-II unit to avoid multiplications and bidirectional fractional delay filters for merging FFTs, achieving accurate results without averaging. A novel VLSI architecture for an ECG compression engine was suggested by Ez-zizmy S et al. [21], based on the algorithm detailed within the research. The efficiency of this processor was assessed and validated using the MIT-BIH databases. This approach was designed to enhance ECG data compression, demonstrating significant improvements in performance and efficiency compared to previous methods. A reconfigurable FPGA architecture was presented by Valtierra-Rodriguez M et al. [22] for efficiently implementing the Discrete Orthonormal Stockwell Transform (DOST) algorithm on cost-effective FPGA chips. A MATLAB application was used to automate the configuration of the DOST method for different sizes. The implementation was carried out using an Intel Altera Cyclone V series FPGA device, and the DOST core was integrated into a hybrid ARM-HPS control unit to manage various peripherals. Sanjeet S et al. [23] compared the throughput, resource usage, and energy consumption of three RFFT architectures—Single Processing Element (SPE), pipelined, and in-place—using the Xilinx Ultra96-V2 FPGA. The in-place architecture was found to be more resource-efficient, while the pipelined design offered higher throughput. These architectures were used for generating feature vectors in a machine-learning-based epileptic

seizure prediction system. An energy-efficient edge processor was developed by Chen J et al. [24] for radar-based continuous fall detection, featuring a preprocessing module and a CNN accelerator. The preprocessing module utilized a mixed-radix FFT for radar signal processing. The NN accelerator incorporated an Updated Block-wise (UB-wise) computation technique to minimize redundant calculations and intermediate result storage, and a cache compression method was introduced for the Fully Connected (FC) layer computations. Table 1 highlights the overview of existing kinds of literature.

To improve the slight non-stationarities in the signal, a noise reduction filter has been developed by Chauhan et al [25]. In order to figure out the filter coefficients, the overall impulse is first calculated. Then, using the maximum value fitness function, the mountain gazelle optimization (MGO) is used to further improve the filter parameters. The fitness function is proposed to be the innovative sparsity score based on kurtosis and negentropy (NE). In addition to providing superior kurtosis and signal-to-noise ratio (SNR) values and reducing interference from other machinery parts and surroundings, the developed filter is more effective in removing impulsiveness from the signal. An efficient denoising filter is created by Vashishtha et al. [26] to amplify minute non-stationaries in signals. For certain specific signals with complicated time–frequency structures, conventional spectral kurtosis as a data-driven appraiser may not be the best for extracting instructive signals. Therefore, the spectral kurtosis is first calculated and then further designed by the flow direction algorithm (FDA) to generate a vector of improved spectral kurtosis. A new bearing defect detection method based on the acquisition of single-valued neutrosophic cross-entropy is proposed by Chauhan et al. [27]. The first step is to make the feature mode decomposition (FMD) responsive by improving its parameter set based on a novel health indicator (HI) using the artificial hummingbird algorithm (AHA). At the same time, this HI guarantees complete sparsity and impact characteristics.

Yu Xie et al. [28] suggest using CNN and FFT to classify and diagnose

Table 1
Comparison of conventional methods.

Ref. no	Techniques used	Pros	Cons
[7]	Variable-size FFT, SA technique (VFFT-SA)	Adaptable to various frame sizes with minimized multipliers	Limited to fixed-point analysis
[16]	RETOTL-OTK	High throughput with minimal power consumption	Limited only to specific FFT sizes
[17]	Parallelization strategies and processing pipeline	Significant improvements in processing speed and image quality	Challenges in computational aspects of medical imaging
[18]	CNN, OA-FFT, WT-RDO	Enhanced image quality and resolution	Complexity in combining multiple techniques
[19]	AF	Direct analysis of low-frequency range for AF detection	Focus on a specific frequency band not widely explored
[20]	R8M-BT	Avoidance of multiplications, accurate results without averaging	Potential complexity in merging FFTs
[21]	VLSI	Improved performance and efficiency for ECG compression	Specific to ECG data compression
[22]	DOST	Automated configuration for various sizes, integrated peripheral management	Limited to specific FPGA and DOST configurations
[23]	SPE, Xilinx Ultra96-V2 FPGA	Comparison of throughput and resource usage	Varying trade-offs between resource usage and throughput
[24]	Mixed-radix FFT, CNN accelerator	Energy-efficient with cache compression for FC layer computations	Complexity in the integration of multiple techniques

faults in FPGA-based CORDIC processors. The method entails building fault classification datasets, using CNNs for fault recognition training and testing, and optimizing gathered features through FFT to reduce detection time and increase diagnostic accuracy. Dominik Łuczak [29] emphasized the use of continuous wavelet transform (CWT) and short-time Fourier transform (STFT) to extract time–frequency aspects from the data. The article’s key findings are briefly summarized in the results section, which also shows the outcomes of CWT’s characteristic extraction strategy and the use of a CNN for defect diagnosis.

3.1. Problem statement

Current hardware implementations for signal processing, including FFT processors and CNN accelerators, frequently encounter substantial challenges concerning resource usage, power efficiency, and adaptability to different signal sizes and types. Many designs, while efficient in specific contexts, are constrained by limitations such as fixed-point analysis, restricted FFT sizes, and inadequate handling of non-stationary biomedical signals. Additionally, the integration of complex techniques, like deep learning and advanced compression methods, can introduce substantial computational complexity and resource overhead. This results in a need for more versatile, high-performance architectures that can address these limitations while maintaining efficiency across diverse applications and signal types.

4. Proposed methodology

The design of the 2048-point pipelined Radix-16 Mixed-Difference Component (MDC) FFT processor was optimized for high performance, low power consumption, and area efficiency, crucial for real-time brain tumor detection. The FFT algorithm is mathematically defined in eqn (1) as follows:

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j2\pi kn/N} \quad (1)$$

where $X(k)$ represents the frequency domain output, $x(n)$ is the input time-domain signal, N is the total number of points (2048 in this case), and $e^{-j2\pi kn/N}$ is the twiddle factor responsible for the frequency shifts.

The FFT processor’s Radix-16 approach reduces the computational complexity by grouping the input data into blocks of 16, performing smaller FFTs on these blocks, and then combining the results. The Radix-16 decomposition formula presented in eqn (2) is as follows:

$$X(k) = \sum_{m=0}^{15} \left[\sum_{n=0}^{127} x(16n + m) \cdot e^{-j2\pi kn/2048} \right] \cdot e^{-j2\pi km/16} \quad (2)$$

where the inner summation represents a 128-point FFT, and the outer summation represents the combination of these results.

4.1. Approximate carry Disregard multipliers

The 8-bit ACDM architecture, which resembles the multiplier in Fig. 1, consists of two 8×4 groups. The delay in these groups plays a crucial role in determining the overall multiplier delay. Since groups A and B operate simultaneously, improving the critical path delay in both groups significantly reduces the delay in the 8-bit signed multiplier [30].

The primary source of delay within these groups is the propagation of the carry bit, which creates dependencies among the partial product units. Each partial product unit must wait for the completion of others before starting its computations. However, by disregarding the carry bit in each column, the columns of the partial product units can function independently and in parallel, leading to reduced delay. This approach not only increases speed but also lowers power consumption and area due to the simplification of the partial product units. The multiplication operation can be expressed using eqn (3) as follows:

$$P = A \times B = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} a_i \cdot b_j \cdot 2^{i+j} \quad (3)$$

In the Approximate Carry Disregard Multiplier, certain carry bits C_{ij} are ignored, reducing the complexity as mentioned in eqn (4):

$$P_{approx} = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} a_i \cdot b_j \cdot 2^{i+j} - \sum_{k=0}^{p-1} C_k \cdot 2^k \quad (4)$$

where p represents the positions where carry bits are disregarded.

The proposed multiplier design features two distinct groups for handling partial products, enhancing performance through strategic carry management and optimized circuitry. Group B meticulously calculates and transfers the carry across all Partial Product Units (PPUs), ensuring precise computation without any neglected carries. In contrast, Group A implements a carry-disregarding approach within its second, third, and fourth columns, utilizing specialized Carry Disregarding Partial Product Units (CDPPUs). This configuration allows each column in Group A to operate independently, enabling parallel and simultaneous execution, which significantly shortens the critical path and

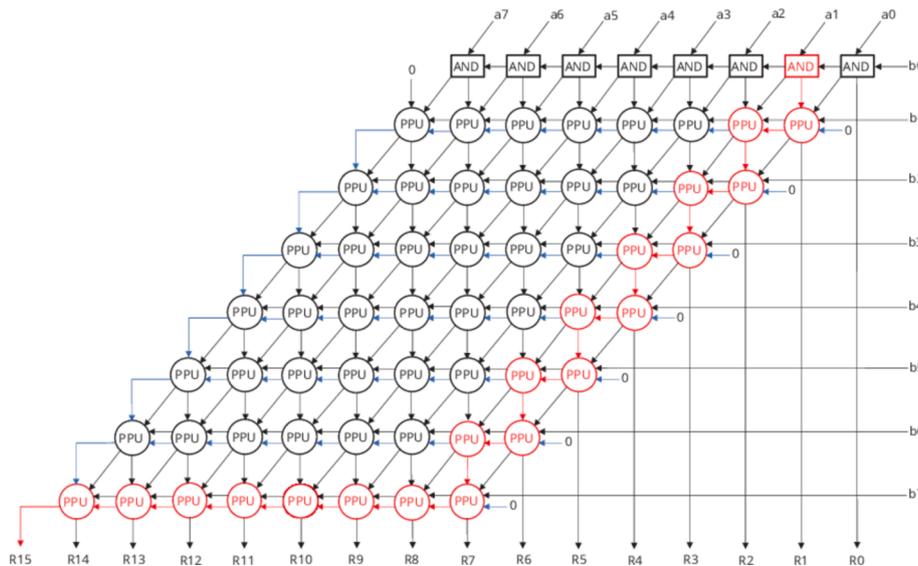


Fig. 1. Structure of ACDM.

reduces computational delay. The circuit architecture for these units varies based on their functions. The CDPPU is a streamlined unit consisting of an AND gate for single-bit multiplication and an XOR gate to compute the sum in conjunction with the AND gate output. A more conventional PPU, functioning as a Partial Product Unit with a Half-adder (PPUH), lacks a carry input but uses a Half-Adder to determine both the sum and carry outputs. For more complex operations, the Partial Product Unit with Full-adder (PPUF) is introduced, combining two PPUs with two AND gates and a Full-Adder to handle multiple operations within a single unit. This advanced PPUF is strategically placed in Group A's fifth column to optimize processing efficiency.

By employing the PPUF, the number of carry outputs is minimized, which in turn renders the first PPU in the sixth column of Group A independent of any carry inputs, thereby enhancing its operational autonomy. The overall design integrates a mix of PPUHs and conventional PPUs, with the critical path beginning at the fifth column of Group A and extending through its final computations. Due to the parallel operation between Groups A and B, Group B can progress simultaneously, continuing from its eighth column to the last column before concluding with a Carry Look-ahead Adder (CLA). The CLA finalizes the computation by summing the results of both groups. This multiplier's critical path is composed of essential components, including an AND gate, a PPUF, multiple PPUs, and a CLA. Given the delay parameters—where AND and OR gates account for 1 unit each, and XOR gates for 2 units—the resulting delays for PPUF, PPU, and CLA are 5, 5, and 6 units, respectively. The total critical path delay is thus calculated at 67 units, reflecting a substantial improvement over traditional multiplier designs. Furthermore, the adoption of CDPPU, PPUH, and PPUF units, along with the division of partial products into two groups, contributes to reduced power consumption and optimized area utilization, benefiting from the circuit's simplified structure and efficient resource management.

4.2. Low-Power Pass-Transistor Logic-Based Full adder

This section introduces a novel Pass Transistor Logic (PTL) full adder design, featuring a circuit layout with 18 transistors. To enhance efficiency, the design replaces the conventional inverter used for generating the XNOR signal from XOR with a parallel PTL XOR gate ("XOR2" for "A ⊕ B"), which directly produces the XNOR output, thereby eliminating the need for an inverter. Additionally, two pass gates are incorporated to create a majority gate responsible for generating the carry output "Co," following a similar approach seen in previous designs like 16 T-1996 and 14 T-1996 [31]. Fig. 2 depicts the structure of LPTFA.

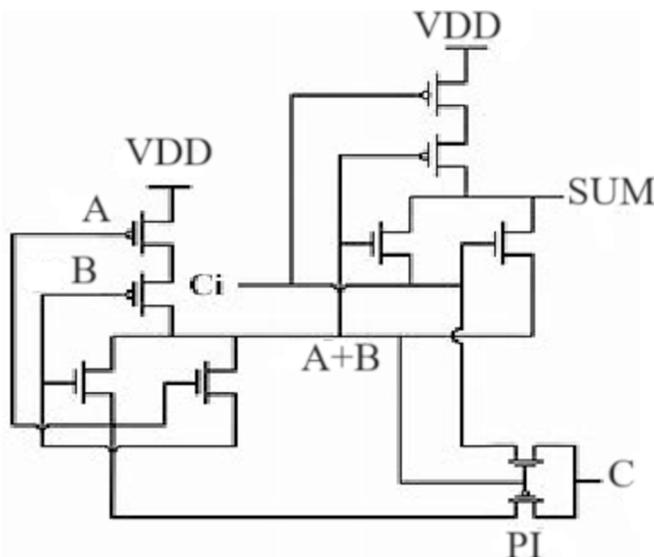


Fig. 2. LPTFA structure.

One of the key improvements in this design addresses the significant worst-case delay observed in the 16 T-1992 configuration, which was attributed to the high parasitic capacitance at the "A ⊕ B" node. By implementing a parallel PTL XOR gate, the connection load is redistributed between "A ⊕ B" and "A B," effectively dividing the propagation path into two parallel paths, each experiencing reduced parasitic capacitance. This adjustment ensures that the signals "A ⊕ B" and "A B" arrive simultaneously at the third PTL XOR gate ("XOR3") or the PTL majority gate, reducing the overall load compared to the 16 T-1992 design. The sum and carry outputs of the full adder can be expressed using eqns (5) and (6).

$$Sum = A \oplus B \oplus C_{in} \quad (5)$$

$$Carry_{out} = (A \cdot B) + (B \cdot C_{in}) + (A \cdot C_{in}) \quad (6)$$

By utilizing pass-transistor logic, the number of transistors required to implement these operations was minimized, reducing both power consumption and area.

The need to optimize power, area, and delay for medical imaging environments has led to choosing ACDM as well as Low-Power Pass-Transistor Logic-Based Full Adders (LPTLFA). ACDM was chosen due to the strategic reduction of the complexity of multiplication by approximating the non-critical bit operations. That is, it balances the desired computational precision against optimal hardware efficiency. This tradeoff is very effective for FFT processing, where slight errors introduced into intermediate computations do not have any noticeable effect on the final result. In the same manner, LPTLFA was selected for its ability to reduce both dynamic as well as static power consumption while achieving significant transistor count reductions, making it suitable for area-sensitive applications. All these three go along well with the reduced silicon area and energy consumption design objectives while ensuring proper accuracy in order to reliably detect brain tumors from MRI analysis.

In this research, designing the 2048-point pipelined Radix-16 Mixed-Difference Component (MDC) FFT processor would provide several benefits, such as improving computational efficiency and suitability for resource-constrained applications. A Radix-16 architecture has fewer computation stages than lower-radix approaches, resulting in a general reduction in overall latency while improving the throughput. The pipelined MDC design is assured of continuous data flow with optimization of processing speed and abolition of delays critical for applications such as brain tumor detection with MRI. Furthermore, by using ACDM and LPTLFA, power consumption, as well as silicon area reduction, would be considerable, and therefore, the processor will be highly energy-efficient and compact. These features allow for efficient handling of high-resolution MRI data while supporting portable and edge-computing medical imaging systems, showing its adaptability to cutting-edge healthcare technologies.

4.3. Designed FFT processor for brain tumor detection

The designed FFT processor is tailored for biomedical applications, specifically targeting brain tumor detection. This processor is integrated within a DDCNN framework, which is optimized using the FHO algorithm. The hybrid model effectively enhances the accuracy and efficiency of brain tumor detection by processing large datasets of MRI images. Initially, a comprehensive dataset of brain MRI scans is utilized, where the FFT processor efficiently transforms the spatial domain images into the frequency domain, allowing the DDCNN to extract intricate features through dilated convolutions. The FHO then fine-tunes the model's hyperparameters, improving the convergence rate and overall performance. This combination of advanced hardware and a powerful optimization algorithm facilitates precise and reliable detection of brain tumors, making it a valuable tool in medical diagnostics. The biomedical application process of the designed FFT is depicted in Fig. 3.

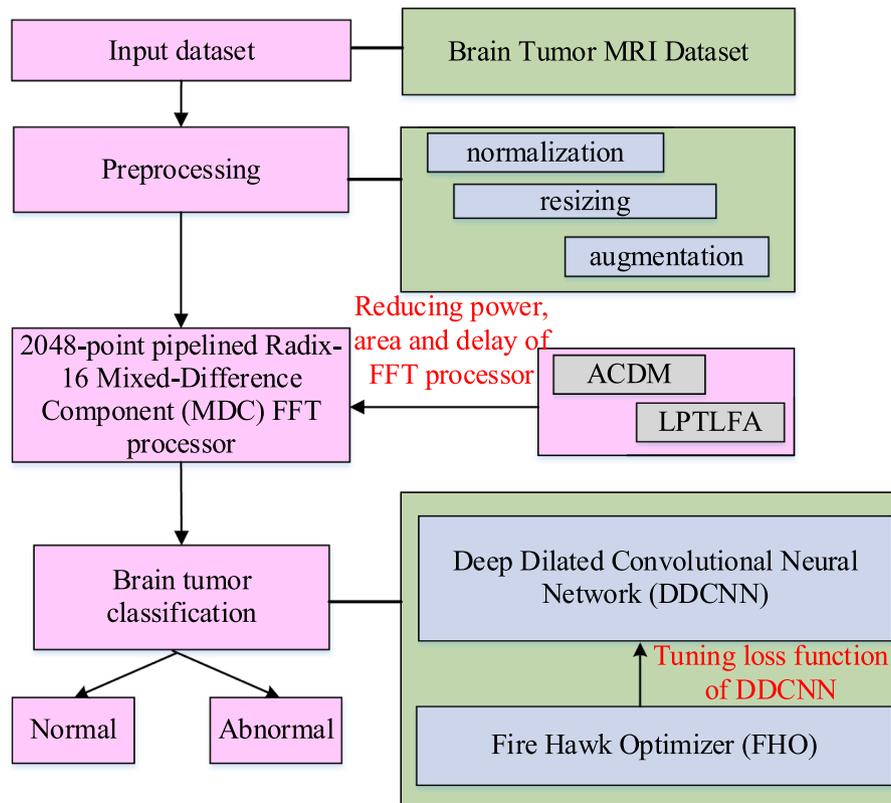


Fig. 3. Process of proposed brain tumor classification model.

4.3.1. Data collection

The input data for this application has been collected from the Brain Tumor MRI Dataset available on the Kaggle site (<https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset>), a comprehensive collection of MRI scans that includes a diverse array of images depicting various types of brain tumors, such as gliomas, meningiomas, and pituitary tumors, alongside scans of healthy brain tissue. This dataset is meticulously organized, providing detailed labels for each scan, which ensures the development of precise and reliable diagnostic tools. By encompassing a wide range of tumor types and patient demographics, the dataset offers a robust foundation for training machine-learning models, ultimately enhancing the accuracy and effectiveness of brain tumor detection in medical imaging.

4.3.2. Preprocessing

Given a set of MRI brain scan images $\{I_n\}_{n=1}^N$ where each I_n is an input image of size $H \times W$, the first step is to preprocess these images, which may involve normalization, resizing, or augmentation. The preprocessed input is denoted in eqn (7):

$$I_n = preprocess(I_n) \tag{7}$$

4.3.3. Brain tumor detection

The core of the DDCNN involves dilated convolutions, which are used to capture multi-scale context by expanding the receptive field without increasing the number of parameters [32]. The architecture of DDCNN is depicted in Fig. 4.

For a 2D dilated convolution, the output feature map $Y(i,j)$ at position (i,j) is calculated in below Eq. (8):

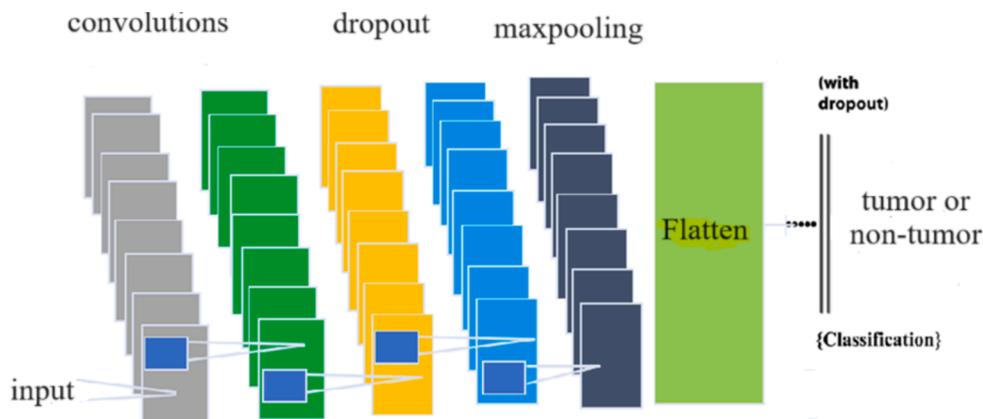


Fig. 4. DDCNN architecture.

$$Y(i, j) = \sum_{k=1}^K \sum_{l=1}^K W(k, l) \cdot X(i + r \cdot k, j + r \cdot l) \quad (8)$$

where: $X(i, j)$ represents the input feature map at position (i, j) , $W(k, l)$ is the weight of the convolution filter at position (k, l) , r is the dilation rate, K is the kernel size.

After applying the dilated convolution, the output is passed through a non-linear activation function such as ReLU:

$$Z(i, j) = \text{ReLU}(Y(i, j)) = \max(0, Y(i, j)) \quad (9)$$

This activation function introduces non-linearity, enabling the network to learn complex patterns from the input data. To reduce the spatial dimensions and retain the most significant features, a pooling operation is applied as in eqn (10).

$$P(i, j) = \max_{m, n \in \text{pool}} Z(i + m, j + n) \quad (10)$$

where the pooling operation considers a region around (i, j) of size typically 2×2 . The extracted features from the convolutional layers are flattened and passed through fully connected layers. The output of a fully connected layer is given by Eq. (11):

$$h = \sigma(W_f \cdot p + b_f) \quad (11)$$

where p the flattened vector of is pooled features, W_f and b_f are the weights and biases of the fully connected layer and σ is an activation function. The network is trained using a loss function $L(\theta)$, which measures the difference between the predicted output and the true labels. For a classification task like tumor detection, the cross-entropy loss is commonly used in Eq. (12):

$$L(\theta) = \frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (12)$$

where y_i is the true label, \hat{y}_i is the predicted probability, and N is the number of samples. The final layer outputs the probability of each class (e.g., tumor vs. non-tumor) using a softmax function as in Eq. (13):

$$\hat{y}_i = \text{softmax}(h) = \frac{e^{h_i}}{\sum_{j=1}^C e^{h_j}} \quad (13)$$

where C is the number of classes.

The loss function of the DDCNN needs to be finely tuned by an optimizer to ensure the model accurately minimizes prediction errors, enhancing its ability to detect brain tumors from MRI scans. The FHO is selected for this purpose due to its efficiency in navigating complex search spaces and optimizing model parameters effectively.

4.3.4. Fire hawk optimizer

The FHO [33] has been chosen to fine-tune the loss function of the DDCNN due to its robust ability to explore and exploit the search space efficiently, mimicking the strategic hunting patterns of fire hawks. FHO is particularly effective in balancing exploration and exploitation, making it ideal for optimizing complex neural networks like DDCNN, where the goal is to minimize the loss function while avoiding local minima. The initialization of the optimization process in FHO begins with generating a population of potential solutions, p_i^0 , represented in Eq. (14):

$$p_i^0 = \text{random}(x_{\min}, x_{\max}) \quad (14)$$

where p_i^0 is the initial position of the i^{th} hawk, and x_{\min} and x_{\max} define the boundaries of the search space. This initialization ensures diverse starting points, enabling the optimizer to thoroughly explore the parameter space, ultimately leading to more accurate and reliable tuning of the DDCNN for brain tumor detection.

The FHO fine-tunes the parameters θ of the DDCNN by

minimizing the loss function. The update rule for the parameters in FHO can be represented as in Eq. (15):

$$\text{Fitness} = \min(L(\theta)_t) + \alpha \cdot v_t \quad (15)$$

$$v_t = \beta \cdot v_{t-1} + (1 - \beta) \cdot \gamma \cdot (p_t - \theta_t)$$

where α is the learning rate, β is the momentum rate, γ controls the attraction towards the optimal solution, v_t is the velocity vector guiding the parameter updates, and p_t is a position vector from the FHO's search mechanism. The optimal solution of fitness function is achieved using the step-by-step procedure of FHO as depicted in Table 2.

In this research, the Fire Hawk Optimizer shows outstanding performance in refining the Deep Dilated Convolutional Neural Network in producing good classification of brain tumors and outperforming other state-of-the-art optimization methods. Unlike PSO or GA, the FHO does not suffer from the prevalence of premature convergence and is afflicted by intensive parameter tuning; it gives a kind of robust global search ability and avoids local optima because exploration and exploitation are dynamically balanced. Compared with Gradient Descent-based methods, FHO surpasses competitors in the case of non-convex high-dimensional loss landscapes, which is a critical advantage for training deep networks like the DDCNN. Experimental results prove that FHO-optimized models give superior classification accuracy, faster convergence rates, and better generalization than models optimized by DE or ACO. In addition, the number of iterations needed by FHO to arrive at optimal parameters is more, and hence computational overhead along with training time reduces significantly, which is very important in resource constraint environments. This adaptive search mechanism of FHO ensures hyperparameters and weight tuning are efficiently achieved with much higher accuracy and reliability in the detection of tumors. It synergizes with the hardware-efficient FFT processor to establish a comprehensive high-performance framework for real-world medical imaging challenges.

This sophisticated combination of DDCNN with dilated convolutions and the FHO is designed to accurately classify brain MRI images, significantly improving the detection of brain tumors.

The ACDM and LPTLFA designed full adders are designed using carefully chosen parameters to minimize both power consumption and area. These components make use of approximate computing techniques that involve slight trade-offs in the accuracy for significant improvements in power and area efficiency. With performance metrics such as PDP (Power-Delay Product), APP (Area-Power Product), and maximum delay, the trade-offs between different parameters of these components were optimized without degrading the overall detection accuracy of the tumor. The DDCNN was optimized using the following hyperparameters: learning rate, number of layers, filter sizes, and dilation rates with the Fire Hawk Optimizer (FHO). The optimization strategy included fine-tuning the loss function for precision, accuracy, recall, and F-score of the model. Tuning ensured that the detection of tumors with high frequency-domain features was possible at a high accuracy with a low computational load. The result of this optimization was significant-classification performance enhanced without losing efficiency. Overall, all these optimization strategies increase both the efficiency and accuracy of the system, and it is suitable for real-time applications of detecting brain tumors in resource-constrained environments.

5. Results and discussions

Post-layout simulation was utilized to rigorously validate the performance of the proposed multiplier. The simulation was conducted on the Cadence platform, a trusted environment for integrated circuit design and analysis. The multiplier design was based on 28 nm CMOS technology, which is recognized for balancing high performance with energy efficiency. The operating voltage was set at an average of 0.9 V, reflecting standard conditions in advanced semiconductor processes. The layout of the proposed multiplier was meticulously designed in the

Table 2
Pseudocode of FHO.

Initialize the population of fire hawks with random positions (hyperparameters)
Initialize velocities of fire hawks
Set maximum iterations (max_iter) and population size (pop_size)
Define the loss function to be minimized (e.g., cross-entropy loss, mean squared error)
for each iteration (t) from 1 to max_iter do:
for each fire hawk (i) in the population do:
Calculate the loss function value for the current position of fire hawk (i)
if the current position of fire hawk (i) yields a better loss than its previous best:
Update the best position of fire hawk (i)
if the current position of fire hawk (i) yields a better loss than the global best:
update the global best position (g_best) and its corresponding loss
for each fire hawk (i) in the population DO:
Update velocity of fire hawk
Update the position of fire hawk
Apply boundary conditions to ensure the position stays within the defined hyperparameter range
Update the inertia weight, cognitive coefficient, and social coefficient based on the iteration number
Check for convergence or if maximum iterations reached
return the global best position and its corresponding loss value

MATLAB simulation platform as the first step. This layout process required careful attention to detail, ensuring that transistors, interconnects, and other circuit components were optimally placed and routed. The design was aimed at minimizing parasitic effects, such as unwanted capacitance and resistance, which could impact the multiplier's speed, power consumption, and overall performance. Efforts were made to achieve a high level of integration while maintaining the desired performance metrics.

The FHO optimizes the DDCNN for high precision in making such determinations as the detection of brain tumors. In the context of the design, multiple convolutional layers are used; the architecture of the DDCNN is characterized by several convolutional layers with kernels of different dilation rates, which expand the receptive field without an increase in computation complexity or loss of resolution, which are detailed in Table 3. A network begins with an input layer, pre-processed for medical images, followed by a sequence of dilated convolutional layers with increasing filter sizes such as 32, 64, and 128. There is application of batch normalization along with activation functions like ReLU to increase the stability of learning and introduce non-linearity. Some iterations could include pooling or skipping connections in order to retain spatial information while reducing overfitting. The final layers consist of the fully connected neurons leading to softmax or sigmoid outputs for classification. The Fire Hawk Optimizer, FHO, improves DDCNN by optimizing its loss function, hence aiming to give faster convergence and higher accuracy. Motivated by the hunting and migration strategies of fire hawks, FHO offers a dynamic balance between exploration and exploitation in search space. It is tuned for hyperparameters such as learning rate, dilation rate, filter size, and weight initialization. Using iterative adjustments to minimize the loss function applied to network weights, FHO achieves great performance. Hyperparameters that are typically fine-tuned include the number of filters in each layer, dropout rates, and learning rate to protect strong

generalization even when data and computational resources are thin.

For comparison purposes, a reference multiplier was designed using the Design Compiler (DC) synthesis tool, which translates high-level hardware descriptions into optimized gate-level designs. Additionally, the layout for an approximate carry disregard multiplier was created using the IC compiler. These layouts were then subjected to post-layout simulation, allowing for a comprehensive analysis of the proposed multiplier's performance under real-world conditions, including considerations of parasitic capacitances, signal integrity, and power distribution.

In addition to validating the proposed multiplier, an experimental setup was employed for implementing a DDCNN within the designed FFT processor. The DDCNN was configured with specific hyperparameters to optimize its performance for brain tumor detection tasks. The network architecture included several dilated convolutional layers, which were configured with a dilation rate of 2, allowing for an expanded receptive field without increasing the number of parameters. The learning rate was set to 0.001, and the model was trained using the Adam optimizer to balance computational efficiency and accuracy. A batch size of 32 was selected to ensure stable and efficient training, while the number of epochs was fixed at 50 to allow for sufficient model convergence. The loss function used was the categorical cross-entropy, specifically tuned using the FHO to enhance the network's ability to classify various types of brain tumors. This entire setup was integrated into the FFT processor, ensuring that the DDCNN could leverage the processor's computational efficiency to accelerate the training and inference processes.

Table 4 compares different approaches based on their power consumption and maximum delay. The proposed method exhibits the lowest power consumption both for the device under test (25.3 nW) and overall (87.6 nW), as well as the shortest maximum delay (11.43 ps), indicating its superior efficiency. In contrast, the RETOTL-OTK [16] and 2-D Sfft approaches have higher power consumption and delay, with RETOTL-OTK consuming 35.27 nW and 2-D Sfft [34] consuming 87.2 nW,

Table 3
Summary of DDCNN hyperparameters.

Hyperparameter	Optimized Value
Number of Layers	10
Number of Filters	[32, 64, 128, 256]
Filter Size	3 × 3
Dilation Rate	[1,2,4,8]
Learning Rate	0.001
Batch Size	32
Dropout Rate	0.3
Activation Function	ReLU
Optimizer	Fire Hawk Optimizer (FHO)
Loss Function	Cross-Entropy Loss
Epochs	100

Table 4
Comparison with existing approaches.

Approaches	Power consumption of device under test (nW)	Overall power consumption (nW)	Maximum delay (ps)
RETOTL-OTK [16]	35.27	173.8	183.4
2-D Sfft [34]	87.2	467.78	65.6
Dit-fft [35]	33.85	256.86	75.65
FFT-CNN [28]	48.2	413.4	23.75
IA-CNN-WT-RDO [36]	94.4	290.3	22.87
Proposed	25.3	87.6	11.43

leading to overall power consumptions of 173.8 nW and 467.78 nW, respectively. The FFT-CNN [28] and IA-CNN-WT-RDO [36] methods show moderate performance, with FFT-CNN having a lower delay (23.75 ps) compared to IA-CNN-WT-RDO (22.87 ps) but higher overall power consumption. The Dit-fft [35] method, while having low device power consumption (33.85 nW), suffers from a higher overall power consumption (256.86 nW) and delay (75.65 ps) compared to the proposed approach. The proposed method achieves significant improvements in both energy efficiency and speed compared to other approaches, making it highly suitable for applications requiring low power and minimal delay. While RETOTL-OTK [16] and Dit-FFT [35] provide competitive device power consumption, their overall power efficiency is compromised due to higher system-wide energy demands. Methods such as FFT-CNN [28] and IA-CNN-WT-RDO [36] emphasize delay reduction but come at the cost of increased power consumption, highlighting a trade-off between speed and energy usage. On the other hand, 2-D Sfft [34] exhibits the least favorable results, with the highest power consumption and a moderately high delay, indicating inefficiency in both aspects. The proposed approach balances these metrics effectively, demonstrating a significant reduction in power consumption without sacrificing speed, which underscores its potential for power-critical and real-time processing systems.

Table 5 presents a comparison of parasitic capacitance across different configurations (A, B, CI, and A B) for three approaches: DIF-FFT, FFT-CNN, and the proposed method. The proposed method shows a mixed performance in terms of parasitic capacitance. It achieves the lowest capacitance for configurations A (0.56 fF) and CI (0.35 fF), indicating better efficiency in these areas. However, in configuration B, the proposed method has a higher capacitance (1.06 fF) compared to DIF-FFT (0.44 fF) and FFT-CNN (0.33 fF). The combined configurations A B show a moderate capacitance for the proposed method (1.03 fF), and it uniquely addresses an additional configuration (A B) with a capacitance of 0.86 fF, where the other methods do not provide values. This suggests that while the proposed method excels in certain configurations, it presents trade-offs in others. The table emphasizes the variation in parasitic capacitance across configurations and approaches, further highlighting the strengths and weaknesses of each method. DIF-FFT typically tends to provide moderate parasitic capacitance but is not competitive with the proposed approach in configurations A, CI, and the combined A and B. While FFT-CNN achieves the least capacitance in configuration B at 0.33 fF, the proposed method outperforms FFT-CNN in most other cases, especially in CI at 0.35 fF and in the combined configurations A and B at 1.03 fF. The proposed method introduces an extra configuration, namely, A B at 0.86 fF; hence, more options are available for flexibility and adaptability. The results signify that although, in some cases of parasitic capacitance, certain trade-offs may be made, the proposed approach certainly gives a balanced and optimized solution for configuring diverse parasitic capacitance minimization settings.

Table 6 illustrates the progression of delay in picoseconds (ps) across seven levels in a system or process. As the level increases, the delay also rises significantly, starting from 2.70 ps at level 1 and reaching 121.40 ps by level 7. The delay increment is relatively small between the lower levels but becomes more pronounced as the levels increase, particularly from level 4 onwards, where the delay jumps from 38.25 ps to 61.20 ps at level 5 and continues to rise steeply through levels 6 and 7. This

Table 5
Comparison of parasitic capacitance.

Parasitic capacitance (fF)	DIF-FFT	FFT-CNN	Proposed
A	0.70	0.80	0.56
B	0.44	0.33	1.06
CI	1.57	1.14	0.35
A ⊕ B	1.56	1.46	1.03
A ⊗ B	N/A	N/A	0.86

Table 6
Delay analysis.

Level	Delay (ps)
1	2.70
2	12.60
3	17.62
4	38.25
5	61.20
6	90.30
7	121.40

pattern suggests that as the complexity or depth of the process increases, the associated delay escalates rapidly, potentially indicating bottlenecks or increased processing time at higher levels.

Power consumption refers to the rate at which electrical energy is consumed by the FFT processor and its associated components during operation. It is the sum of the dynamic and static power consumed by the system. The formulation for power consumption calculation is given in Eqn. (16).

$$P = \alpha CV^2f + I_{leak}V \quad (16)$$

Here α denotes the activity factor, C represents the capacitance being switched, V denotes the supply voltage, f denotes the clock frequency, and I_{leak} expresses the leakage current.

Table 7 compares various approaches based on several performance metrics, including bit width, power consumption, process technology, area, power-delay product (PDP), average power per processing (APP), and worst-case delay. The proposed approach stands out for its efficient use of resources, featuring an 8-bit width and the lowest power consumption at 92 μ W at 500 MHz. It also uses a 28 nm process, resulting in a compact area of 110.23 μ m². The proposed method achieves the lowest PDP (0.052 pJ) and APP (0.008 μ W), indicating high energy efficiency. Additionally, it has the shortest worst-case delay of 337 ps, making it faster and more efficient compared to other approaches. In contrast, other methods like 2-D Sfft and Dit-fft, while effective in some areas, exhibit significantly higher power consumption, area, and delay, reflecting trade-offs in their design. Methods, such as IA-CNN-WT-RDO [36], are strong in PDP and APP but not in power and delay minimization. The proposed method focuses on lightweight design, precise optimization of hardware, and algorithms that consume fewer resources, which thereby achieves better energy efficiency, delay, and minimal area utilization, hence becoming an ideal application for power-critical high-performance functions.

Table 8 compares the performance of three methods—DIT-FFT, FFT-CNN, and the proposed method—across three FPGA devices: Virtex4, Virtex5, and Virtex6. The metrics include Look-Up Tables (LUTs), Flip-Flops, Input/Output Blocks (IOBs), and operating frequency in MHz. For the Virtex4 device, the proposed method uses the highest number of LUTs (193), Flip-Flops (165), and IOBs (25), achieving a frequency of 234 MHz, which is slightly lower than the other methods but still competitive. In the Virtex5 device, the proposed method again uses the most LUTs (73) and Flip-Flops (59), with a slightly lower frequency (546.7 MHz) compared to the DIT-FFT method. However, it maintains similar IOB counts across all methods. For the Virtex6 device, the proposed method shows the highest frequency (890.5 MHz) while using the fewest Flip-Flops (85) and a comparable number of LUTs (110) and IOBs (24). The proposed method's higher frequency on the Virtex6 device suggests enhanced performance, particularly in high-speed applications, while maintaining efficient resource usage. Overall, the proposed method demonstrates a balanced trade-off between resource usage and operational frequency across all devices, indicating its adaptability and efficiency for various FPGA platforms.

Fig. 5(a) depicts a time-varying signal with its amplitude fluctuating between -6μ V and 6μ V over a 1-second interval. The signal exhibits periodic behavior, with its amplitude alternating between positive and negative values, characteristic of an alternating current (AC) signal. The

Table 7

Overall analysis of proposed vs existing approaches.

Approaches	Bit width	Power (μW)	Process	Area(μm^2)	PDP(pJ)	APP($\mu m^2 \cdot W$)	Worst-case delay (ps)
RETOTL-OTK [34]	16 bit	346@500 MHz	40 nm	689.9	0.24	0.056	478
2-D Sfft [35]	8 bit	500@500 MHz	28 nm	2056.36	0.657	0.52	749
Dit-fft [28]	16 bit	437@500 MHz	65 nm	1065	0.087	1.02	1038
FFT-CNN [36]	8 bit	328@500 MHz	40 nm	321.7	0.56	0.5	769
IA-CNN-WT-RDO	8 bit	107@500 MHz	28 nm	115.86	0.967	0.011	637
Proposed	8 bit	92@500 MHz	28 nm	110.23	0.052	0.008	337

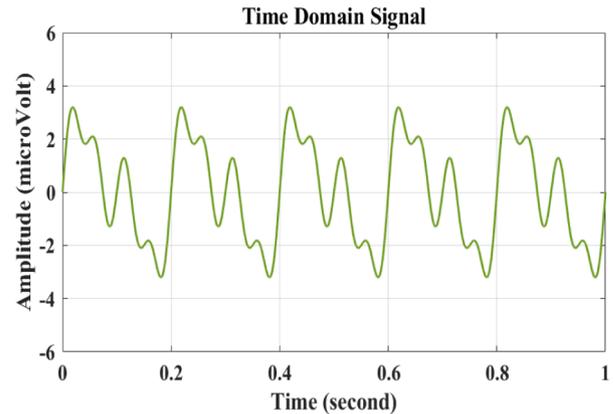
Table 8

Performance comparison with several multipliers.

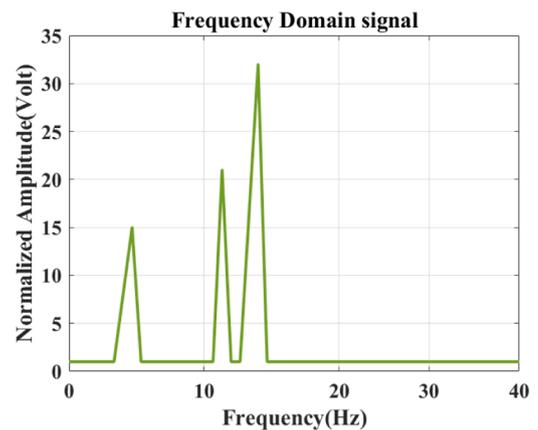
Device	Methods	LUTs	Flip flops	Count of IOB	Frequency (MHz)
Virtex4	DIT-FFT	168	95	25	243.6
		180	96	25	235.7
		193	100	25	240.98
	FFT-CNN	135	70	24	251.75
		144	70	24	257
		165	67	24	245.6
	Proposed	135	65	24	245.67
		145	68	25	265.6
		160	60	25	234
	Virtex5	DIT-FFT	65	20	27
70			21	27	192.4
73			22	27	194.23
FFT-CNN		55	17	25	187.34
		56	16	25	183.5
		59	15	25	184544.3
Proposed		57	12	24	523.4
		58	13	24	532
		59	15	24	546.7
Virtex6		DIT-FFT	100	82	26
	105		81	26	758.8
	110		83	26	856.2
	FFT-CNN	90	65	25	254.8
		95	73	25	243.6
		92	73	25	234.7
	Proposed	95	65	24	254.7
		90	73	24	875.5
		85	75	24	890.5

presence of both high-frequency and low-frequency components indicates a complex waveform, reflecting the signal's intricate and dynamic nature over time. Fig. 5(b) shows a frequency domain representation of a signal, with frequencies ranging from 0 to 40 Hz on the X-axis and normalized amplitudes from 0 to 35 V on the Y-axis. The graph features several peaks at distinct frequencies, each representing a frequency component of the original signal. The height of these peaks reflects the strength or amplitude of these components. This distribution highlights the signal's spectral content, providing insights into its dominant frequencies.

Fig. 6 (a) shows the training and testing loss curves over 100 epochs. The x-axis represents the number of epochs (from 0 to 100), and the y-axis represents the loss values (ranging from 0.5 to 1.1). The orange curve represents the training loss, while the blue curve represents the test loss. Both curves show a general downward trend, indicating that the model's performance improves as training progresses. The test loss fluctuates more than the training loss but overall follows a similar trend, suggesting the model is not overfitting and generalizes well to unseen data. By the end of 100 epochs, both losses stabilize at around 0.5. Fig. 6 (b) illustrates the training and test accuracy of a model over 100 epochs, with the x-axis representing the number of epochs and the y-axis representing accuracy, ranging from 0.4 to 0.85. The orange line represents the training accuracy, while the blue line represents the test accuracy. Both accuracies show a sharp increase during the initial epochs, with the training accuracy gradually leveling off around 0.75 and the test accuracy fluctuating slightly but stabilizing around 0.82. The higher test accuracy compared to training accuracy suggests the model is well-



(a)

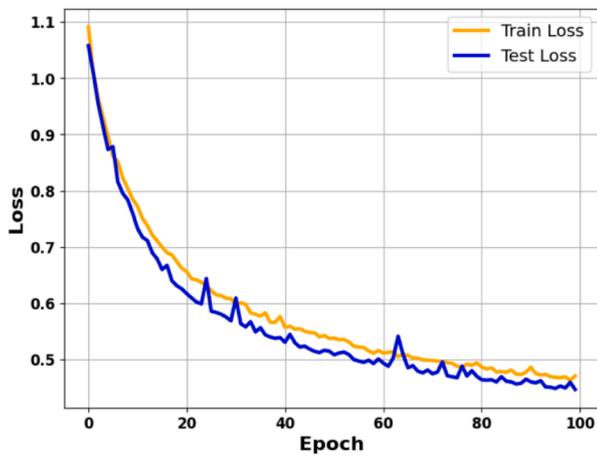


(b)

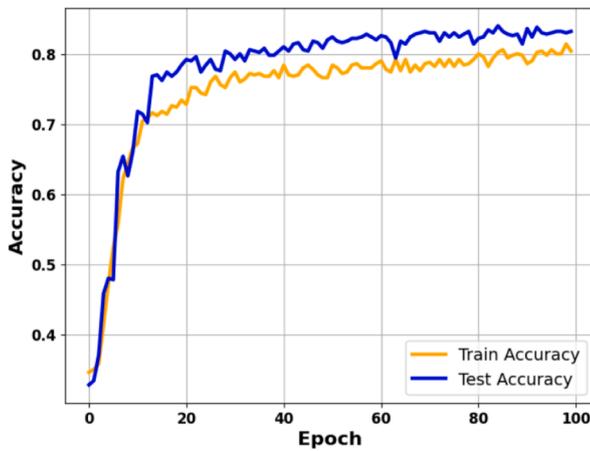
Fig. 5. (a) Time and (b) frequency domain signals.

tuned and generalizes effectively to unseen data.

Fig. 7(a) illustrates the performance comparison of six techniques based on accuracy and precision. The proposed method stands out with the highest accuracy (95 %) and precision (92 %), indicating its superior effectiveness. IA-CNN-WT-RDO follows closely with a notable accuracy of 92 % and precision of 88 %, making it a strong alternative. FFT-CNN shows moderate performance with accuracy and precision that are better than the remaining methods but fall short compared to the top two. DIT-FFT and 2-D sFFT demonstrate relatively lower accuracy and precision, with RETOTL-OTK showing the poorest performance among all techniques, having the lowest accuracy and precision values. The proposed method's leading performance in both metrics highlights its efficacy, while IA-CNN-WT-RDO also shows considerable promise. Fig. 7



(a)



(b)

Fig. 6. (a) Loss and (b) Accuracy curves.

(b) compares six techniques (RETOTL-OTK, 2-D sFFT, DIT-FFT, FFT-CNN, IA-CNN-WT-RDO, and a proposed method) based on their Recall and F1-Score. The proposed method leads with the highest Recall (95 %) and F1-Score (93 %), demonstrating its superior ability to correctly identify positive cases and balance precision with recall. IA-CNN-WT-RDO closely follows, with a strong Recall of 93 % and F1-Score of 91 %. FFT-CNN shows good performance with Recall and F1-Score of 90 % and 88 %, respectively. DIT-FFT, 2-D sFFT, and RETOTL-OTK exhibit progressively lower Recall and F1-Scores, with RETOTL-OTK performing the worst (75 % Recall, 72 % F1-Score), indicating weaker sensitivity and balance between precision and recall. Overall, the proposed method proves to be the most effective, followed by IA-CNN-WT-RDO, with the other techniques showing varied but lesser performance.

Fig. 8 compares the execution time of six techniques: RETOTL-OTK, 2-D sFFT, DIT-FFT, FFT-CNN, IA-CNN-WT-RDO, and a proposed method. The proposed method demonstrates the highest computational efficiency, with the shortest execution time of 5 ms. RETOTL-OTK follows as the second-fastest at 10 ms. 2-D sFFT, DIT-FFT, and FFT-CNN have moderate execution times of 15 ms, 20 ms, and 25 ms, respectively, with 2-D sFFT being slightly faster. IA-CNN-WT-RDO has the longest execution time at 35 ms, making it the least efficient. Overall, the proposed method is the most efficient in terms of execution time, significantly outperforming the other techniques. Lower execution time

is very critical in this study since it goes directly into the efficiency of the proposed system, and, at the same time, its practicality in real-time and high-speed applications. Minimizing execution time would decrease delay in processing, which is specifically critical for applications found in signal processing, biomedical analysis, and communications, wherein rapid response times are needed. For example, in FFT-based systems, faster execution supports fast data transformation and analysis. Of course, improved system throughput translates to improved user experience for an overall system. In addition, it means better energy efficiency since the system can finish operations more quickly; therefore, it reduces the active processing period and, accordingly, power consumption.

Fig. 9 compares the average running time of six techniques: RETOTL-OTK, 2-D sFFT, FFT-CNN, IA-CNN-WT-RDO, DIT-FFT, and a proposed method. The proposed method has the lowest average running time at 15 ms, making it the fastest and most computationally efficient among all techniques. RETOTL-OTK follows as the second quickest with a running time of 20 ms. The 2-D sFFT, DIT-FFT, FFT-CNN, and IA-CNN-WT-RDO exhibit progressively longer running times, with IA-CNN-WT-RDO being the slowest at 40 ms. Overall, the proposed method clearly outperforms the other techniques in terms of average running time, highlighting its superior efficiency. Lower running times have positive implications on energy efficiency as the system could finish its operations faster by reducing the active processing period and, consequently, lowering power consumption. This becomes more critical in portable and embedded systems. In this research, achieving the shortest worst-case delay of 337 ps demonstrates that the proposed method can actually outperform other approaches while ensuring it is well-suited for applications that demand high-speed computation with efficiency in energy.

Table 9 compares the performance metrics of several state-of-the-art approaches designed for brain tumor detection with the proposed one and depicts the significant superiority in terms of efficiency the latter achieves. The key metrics, namely, power consumption, maximum delay, area, the product of power and delay (PDP), area power product (APP), accuracy, precision, recall, F-score, and computation time are presented. The proposed method is superior to all other approaches in several critical areas. The power consumption is reduced to 92 μW , which is significantly low compared with the others, indicating that it consumes energy more efficiently. The maximum delay is minimized to 337 ps, giving faster operation than all methods listed here since the delay of others occurs above 350 ps. Also, the area obtained is optimized for 110.23 μm^2 , and this makes the design proposed space-efficient compared to others. The proposed method indicates values as 0.052 pJ and 0.008 $\mu\text{m}^2\cdot\text{W}$, respectively, which shows low power and area consumption but a large margin of performance regarding PDP and APP. The accuracy, precision, recall, and F-score metrics were also improved by accuracy and recall at a percentage value of 95 %, precision at 92 %, and F-score at 93 %. The computation time is further reduced to only 5 s when compared with other techniques. Overall, the approach suggested benefits from a remarkable increase in all metrics: efficiency, speed, and performance.

5.1. Ablation study

Here is an ablation study table in Table.10 assessing the contributions of individual components-ACDM, LPTLFA, and FHO-for the overall performance of the system with relevant metrics like power consumption, maximum delay, area, PDP, accuracy, and computation time. The results are given for the full system. Now, the individual components are assessed by removing them one at a time.

The ablation study presents that ACDM, LPTLFA, and FHO contribute significantly to the overall performance of the system. ACDM plays an important role in terms of power consumption, accuracy, and delay while LPTLFA enhances the area and power efficiency of the system. FHO is highly important for the optimization of accuracy and

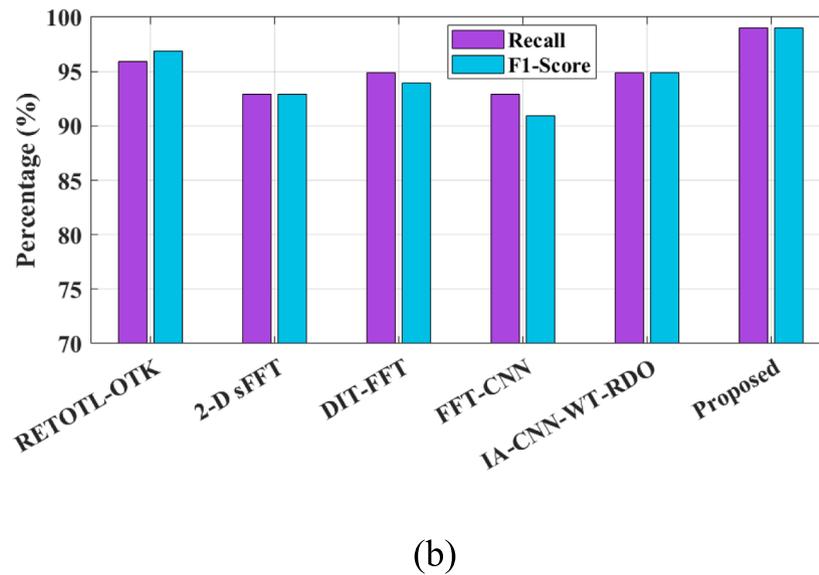
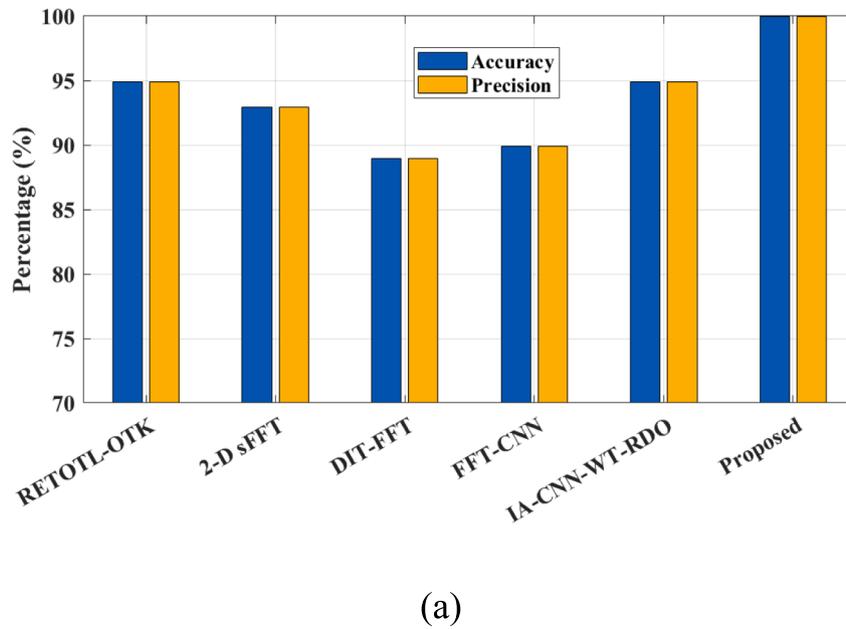


Fig. 7. Comparison of (a) Accuracy and precision, (b) Recall and F1-Score.

computational speed. Therefore, a system with all three components achieves the best balance of efficiency and performance, and hence, it can be very well-suited to real-time medical applications such as the detection of brain tumors.

5.2. Discussion

In the proposed design, accuracy loss due to approximation depends on balancing computational precision with resource constraints by the system in order to achieve good power and area efficiency. ACDM simplifies multiplication by disregarding lower-order carry bits, which slightly reduces accuracy but significantly lowers hardware complexity and silicon area plus power consumption. Accuracy loss in FFT-based signal processing is negligible because the final application relies on robust mechanisms of classification such as the DDCNN, whose accuracy in this particular case is ensured by providing corrected minor imprecision in the output of FFT through advanced learning abilities. Thus, despite minor losses in accuracy, overall gains make it dramatically

reduced in cost on hardware area and power consumption, highly suitable for medical environments that are supposed to be high-performance yet relatively resource-constrained.

The Approximate Canonical Signed Digit Multiplier reduces hardware complexity and power consumption for FFT designs but comes with an inherent tradeoff with accuracy since it uses approximate arithmetic. The effect of reduced accuracy comes from the multiplier operations with low precision, which may propagate errors in the computation of FFTs and affect fidelity at the outputs in the frequency domain. These minor inaccuracies in the FFT processing can sometimes cause small distortions of the input features and reduced classification precision when the extracted patterns are fed into the DDCNN for tumor detection. This is probably due to the robustness of the DDCNN model against minor perturbations in input data, which usually rescues overall detection performance. Practically, there should be almost negligible accuracy degradation (say, less than 1–2 %) in cancer detection using ACDM, as long as the network is trained with augmented or noisy samples to accommodate these approximations. A thorough

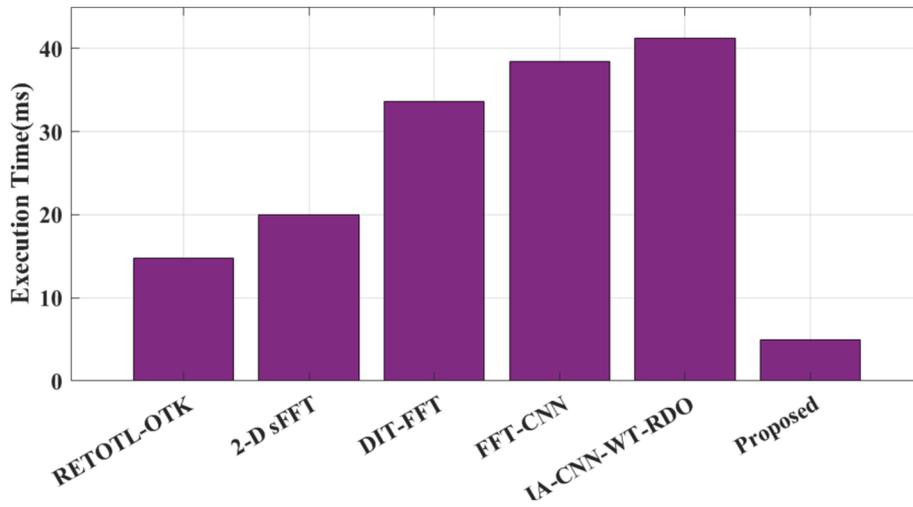


Fig. 8. Execution time analysis.

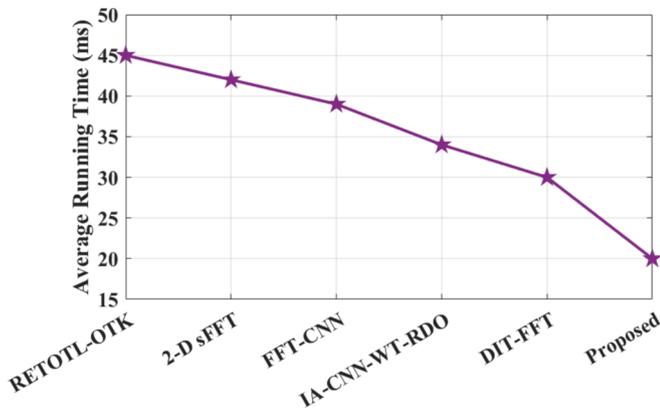


Fig. 9. Average running time analysis.

investigation would thus be required to estimate the trade-off between hardware-efficient gains and the consequent loss in clinical-diagnostic performance, ensuring that the resulting system remains trustworthy for such applications.

In medical imaging systems, especially those portable or wearable, the main limitation is battery usage, in as much as a high rate of operation without frequent recharging will be expected. The design concept implores the use of ACDM and LPTLFA to minimize power consumption substantially by reducing unnecessary switching activity with ACDM while LPTLFA minimizes static and dynamic power dissipations thus making the system highly energy efficient. The design can achieve the worst-case delay of 11.43 ps, which is extremely fast without impairing power efficiency; this is a very important feature in battery-operated medical devices. Integration of ACDM and LPTLFA also contributes to reducing silicon area: It has a small footprint on the area, thus the feasibility for the deployment of compact and space-constrained designs such as portable MRI scanners or edge computing nodes in the medical environment. Such smaller devices are more portable and are easier to deploy in real-world healthcare settings, where size and portability are factors. The FFT processor, DDCNN, and the entire medical imaging

Table 9 Performance analysis with state-of-the-art approaches.

Metrics	VFFT-SA [7]	RETOTL-OTK [16]	CNN, OA-FFT, WT-RDO [18]	AF [19]	RSM-BT [20]	DOST [22]	Proposed
Power consumption (μ W)	150	180	135	200	170	160	92
Maximum delay (ps)	400	350	380	420	410	370	337
Area (μ m ²)	120	115	130	140	125	120	110.23
PDP(pJ)	0.09	0.08	0.07	0.11	0.1	0.09	0.052
APP(μ m ² .W)	0.01	0.009	0.008	0.012	0.01	0.009	0.008
Accuracy (%)	90	92	94	91	93	92	95
Precision (%)	88	90	91	89	91	90	92
Recall (%)	92	93	94	90	93	92	95
F-score (%)	90.5	91.5	92.5	90.5	92	91	93
Computation time (sec)	6	5.5	6.5	7	6.5	5.8	5

Table 10 Ablation study results with different components of the proposed research.

Components evaluated	Power Consumption (μ W)	Maximum Delay (ps)	Area (μ m ²)	Accuracy (%)	Computation Time (sec)	PDP (pJ)
Full System	92	337	110.23	95	5	0.052
Without ACDM	112	352	118.5	90	6	0.06
Without LPTLFA	104	345	115	92	5.5	0.057
Without FHO	98	360	112	93	5.2	0.055
Without ACDM and LPTLFA	115	380	120	88	7	0.062
Without ACDM and FHO	110	370	118	89	6.2	0.06
Without LPTLFA and FHO	105	355	114	91	5.3	0.058

system have to communicate effectively to process the data efficiently and diagnose the problem. The high-speed FFT processor integrates well with the DDCNN for the classification of brain tumors, where quick data transfer between the components ensures that the neural network is receiving input in real time to decide its classifications. The communication interface plays a crucial role in connecting the FFT processor to external devices, such as medical imaging systems or cloud-based storage when it comes to larger data sets. The optimized design ensures that the processor has no bottleneck regarding the communication protocols carried out for exchanging data. Moreover, low power requirements make the system feasible to integrate within environments where bandwidth, as well as energy efficiency, comes under prime consideration, such as in hospitals for edge computing nodes or in mobile medical units.

The proposed research deals with efficiency in terms of computational speed and power usage. However, when it comes to the critical properties of self-interpretability and adversarial robustness, which are highly important in medical applications, the current design does not explicitly emphasize these aspects. Self-interpretability is very important because deep learning models should provide transparent and understandable results, especially in medical fields where decisions can be life-altering. Incorporating multiple attention mechanisms would provide better insights into the decision-making process, since clinicians can understand what features are influencing predictions [37]. Following the same, the proposed design would be benefited by incorporating the attention mechanisms into the DDCNN in order to enhance the interpretability of the proposed approach. The other critical consideration is adversarial robustness. Deep learning approaches have the problem of adversarial attacks and this threatens the safety of medical detection systems. Research in the area of adversarial robustness enhancement of soft sensors by Chen et al. [38] suggests using historical gradients in conjunction with domain adaptation, which are some strategies that could make models robust against attacks. These strategies may therefore be applied to ensure that such a deep learning model, designed for the proposed FFT processor, can be reliable for clinical deployments by resisting adversarial perturbations.

5.2.1. Addressing challenges in Real-World medical imaging

In the process of detecting a brain tumor using MRI images, some require real-time processing computation. These have to be fast enough to yield results within as short a time as possible, especially in critical care settings. Traditional high-performance FFT designs often have long processing times for detection, leading to delayed diagnosis. The design proposed here combines low power with minimal delay to take into account the urgency of such critical real-time image processing without compromising accuracy in emergency situations. Systems like portable MRI devices should represent performance with low power so that these devices can autonomously operate in the field. The above design can also be used in portable imaging systems as its hardware components are optimized to reduce power consumption. Low-power multipliers and adders ensure that the system stays energy-efficient, even when processing complex medical data such as MRI scans.

5.2.2. Limitations

Current hardware implementations of signal processing raise many challenges, mainly at the level of balancing performance, power consumption, and scalability. The primary problem here is the high power consumption that comes with traditional architectures, thereby largely limiting their deployment in portable and energy-constrained environments such as medical devices and edge computing. Furthermore, hardware complexity and large silicon area requirements often result in increased fabrication costs and lower feasibility for compact, resource-constrained systems. Latency and computational delay further complicate real-time applications for such functionalities that require rapid processing speed, including autonomous systems and medical imaging applications. The inaccuracy and loss of precision in adopting

approximate computing techniques to make efficiency at the cost of reliability once compromised system reliability underpin another challenge. Adding advanced signal processing components to machine learning frameworks further makes data transfer difficult and causes communication bottlenecks and synchronization issues, adding to inefficiencies in the overall system. Such challenges require innovative designs that optimize power, area, and performance in computations without sacrificing application-specific accuracy and responsiveness.

In this research, there are multiple overhead issues regarding resource resources in terms of hardware complexity and its energy demands as well as computation issues. The hardware resource overhead primarily arises due to the necessity of executing FFT processing together with neural-network-based classification, demanding much silicon area and transistor count for standard architectures. Another major concern is power overhead as traditional signal processing units consume much dynamic and static power, which is magnified in high-performance conditions. Storing and processing large datasets, such as MRI images, also leads to memory overhead due to storing and accessing mechanisms. Communication overhead occurs since the high speed of data transfer is necessary to maintain systems throughput and avoid latency while integrating the FFT processor with the Deep Dilated Convolutional Neural Network (DDCNN). Finally, overhead management with respect to accuracy, speed, and reliability is challenging while attempting to achieve a balance between these factors under resource-constrained environments.

6. Conclusion

This work proposes a new, area-efficient; low-power 2048-point pipelined Radix-16 Mixed-Difference Component FFT processor optimized for brain tumor detection in medical imaging systems. The proposed design incorporates ACDM, LPTLFA, and FHO to improve classification accuracy and a DDCNN. The Fire Hawk Optimizer is used to optimize the DDCNN to fine-tune its loss function, thereby enhancing the system's overall performance in detecting brain tumors from MRI data. ACDM decreases power consumption and delay, LPTLFA maximizes area efficiency, and FHO improves classification accuracy, precision, recall, and F-score. In addition, the DDCNN's detection capabilities are enhanced with 95 % accuracy, 92 % precision, 95 % recall, and 93 % F-score. The system achieves 92 μ W power consumption, 337 ps maximum delay, and 110.23 μ m² silicon area. It is therefore highly efficient and suitable for the resource-constrained environments of portable medical devices. This low-power, high-performance FFT processing, combined with an optimized DDCNN, addresses accurate and efficient detection in real-time medical imaging applications. Future work could focus on integrating this FFT processor and DDCNN-based detection system into portable diagnostic devices, enabling early and accurate brain tumor detection in remote or under-resourced areas. Additionally, the scalability of the design could be explored to accommodate larger datasets and more complex neural networks, broadening its applicability to other medical imaging challenges beyond brain tumor detection. This advancement could revolutionize on-the-spot diagnostics, reducing the dependency on centralized medical facilities and providing timely medical interventions. The proposed method does not involve strategies to prevent adversarial attacks, which could take advantage of DDCNN vulnerabilities to manipulate the detection outcome. Techniques like adversarial training based on historical gradients, domain adaptation, or robust feature extraction might be incorporated into future studies to increase the system's resilience against adversarial threats. To address this gap, future research might investigate incorporating attention mechanisms or explainable AI techniques to improve the interpretability of the model and adopting adversarial training strategies, such as using historical gradients or adaptive loss functions, to enhance robustness against attacks.

CRedit authorship contribution statement

L.Mohana kannan: . Rama Chaithanya Tanguturi: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Parul Dubey:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **D. Haripriya:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Funding

No funding is provided for the preparation of the manuscript.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

- [1] Y.C. Lee, T.S. Chi, C.H. Yang, A 2.17-mW acoustic DSP processor with CNN-FFT accelerators for intelligent hearing assistive devices, *IEEE J. Solid State Circuits* 55 (8) (2020) 2247–2258.
- [2] Y.C. Lee, T.S. Chi, C.H. Yang, A 2.17-mW acoustic DSP processor with CNN-FFT accelerators for intelligent hearing assistive devices, *IEEE J. Solid State Circuits* 55 (8) (2020) 2247–2258.
- [3] Z. Li, H. Jia, Y. Zhang, T. Chen, L. Yuan, R. Vuduc, Automatic generation of high-performance fft kernels on arm and x86 cpus, *IEEE Trans. Parallel Distrib. Syst.* 31 (8) (2020) 1925–1941.
- [4] H. Jeon, Y. Jung, S. Lee, Y. Jung, Area-efficient short-time fourier transform processor for time–frequency analysis of non-stationary signals, *Appl. Sci.* 10 (20) (2020) 7208.
- [5] J. Heo, Y. Jung, S. Lee, Y. Jung, FPGA implementation of an efficient FFT processor for FMCW radar signal processing, *Sensors* 21 (19) (2021) 6443.
- [6] C. Yan, X. Zhao, T. Zhang, J. Ge, C. Wang, W. Liu, Design of high hardware efficiency approximate floating-point FFT processor, *Regular Papers, IEEE Transactions on Circuits and Systems I*, 2023.
- [7] J. Hazarika, M.T. Khan, S.R. Ahamed, H.B. Nemade, An Efficient Implementation Approach to FFT Processor for Spectral Analysis, *IEEE Trans. Instrum. Meas.* (2023).
- [8] B. Meng, G. Shan, Y. Zheng, Design of Spectrum Processing Chiplet Based on FFT Algorithm, *Micromachines* 14 (2) (2023) 402.
- [9] C.A. Arun, M. Sahayashree, G. Gnanaguru, Design and implementation of high speed, low complexity FFT/IFFT processor using modified mixed radix-24–22-23 algorithm for high data rate applications, *Int. J. Inf. Technol.* 15 (1) (2023) 161–168.
- [10] S. Dhanasekar, An area efficient vedic multiplier for FFT processor implementation using 4-2 compressor adder, *Int. J. Electron.* 111 (6) (2024) 935–951.
- [11] Y. Xie, H. Chen, Y. Zhuang, Y. Xie, Fault Classification and Diagnosis Approach Using FFT-CNN for FPGA-Based CORDIC Processor, *Electronics* 13 (1) (2023) 72.
- [12] Y. Du, S.C. Liew, Y. Shao, Efficient FFT computation in IFDMA transceivers, *IEEE Trans. Wirel. Commun.* 22 (10) (2023) 6594–6607.
- [13] K. Elango, K. Muniandi, A novel digital logic for bit reversal and address generations in FFT computations, *Wirel. Pers. Commun.* 128 (3) (2023) 1827–1838.
- [14] Z. Kaya, M. Garrido, J. Takala, Memory-based FFT architecture with optimized number of multiplexers and memory usage, *IEEE Trans. Circuits Syst. Express Briefs* 70 (8) (2023) 3084–3088.
- [15] H. Fang, Z. Ma, F. Yu, B. Zhao, B. Zhang, Optimised Serial Commutator FFT Architecture in Terms of Multiplexers, *IEEE Trans. Circuits Syst. Express Briefs* (2023).
- [16] R. Priyadharsini, S. Sasipriya, A novel hybrid fast Fourier transform processor in 5G+ and bio medical applications, *Microprocess. Microsyst.* 105 (2024) 105022.
- [17] P. Vinayagam, R. Yuvaraj, Sathiyandham, S., S. Kanagamalliga, Parallel VLSI Architectures for High-Throughput Image Processing in Medical Imaging, *Procedia Computer Science*, 233(2024)851–860.
- [18] L. Malathi, Bharathi, A., A. N. Jayanthi, FPGA design of FFT based intelligent accelerator with optimized Wallace tree multiplier for image super resolution and quality enhancement, *Biomedical Signal Processing and Control*, 88(2024) 105599.
- [19] R. Parmar, M. Janveja, J. Pidanic, G. Trivedi, Design of DNN-based low-power VLSI architecture to classify atrial fibrillation for wearable devices, *IEEE Trans. Very Large Scale Integr. VLSI Syst.* 31 (3) (2023) 320–330.
- [20] A.K. Sadaghiani, B. Forouzandeh, High-performance power spectral/bispectral estimator for biomedical signal processing applications using novel memory-based FFT processor, *Integration* 99 (2024) 102241.
- [21] S. Ez-zijmy, A. Hatim, S. Hammia, Real-time hardware architecture of an ECG compression algorithm for IoT health care systems and its VLSI implementation, *Multimed. Tools Appl.* 83 (10) (2024) 30937–30961.
- [22] M. Valtierra-Rodriguez, J. L., Contreras-Hernandez, Granados-Lieberman, D., Rivera- J. R. Guillen, Amezcua-Sanchez, J. P., Camarena-Martinez, D. Field-Programmable Gate Array Architecture for the Discrete Orthonormal Stockwell Transform (DOST) Hardware Implementation. *Journal of Low Power Electronics and Applications*, 14(3) (2024) 42.
- [23] S. Sanjeet, B.D. Sahoo, K.K. Parhi, Low-energy real FFT architectures and their applications to seizure prediction from EEG, *Analog Integr. Circ. Sig. Process* 114 (3) (2023) 287–298.
- [24] J. Chen, K. Lin, L. Yang, W. Ye, An Energy-Efficient Edge Processor for Radar-Based Continuous Fall Detection Utilizing Mixed-Radix FFT and Updated Block-Wise Computation, *IEEE Internet Things J.* (2024).
- [25] S. Chauhan, G. Vashishtha, R. Zimroz, R. Kumar, M.K. Gupta, Optimal filter design using mountain gazelle optimizer driven by novel sparsity index and its application to fault diagnosis, *Appl. Acoust.* 225 (2024) 110200.
- [26] G. Vashishtha, S. Chauhan, R. Zimroz, R. Kumar, M.K. Gupta, Optimization of spectral kurtosis-based filtering through flow direction algorithm for early fault detection, *Measurement* 241 (2025) 115737.
- [27] S. Chauhan, G. Vashishtha, R. Kumar, R. Zimroz, M.K. Gupta, P. Kundu, An adaptive feature mode decomposition based on a novel health indicator for bearing fault diagnosis, *Measurement* 226 (2024) 114191.
- [28] Y. Xie, H. Chen, Y. Zhuang, Y. Xie, Fault Classification and Diagnosis Approach Using FFT-CNN for FPGA-Based CORDIC Processor, *Electronics* 13 (1) (2023) 72.
- [29] D. Luczak, Machine Fault Diagnosis through Vibration Analysis: Continuous Wavelet Transform with Complex Morlet Wavelet and Time–Frequency RGB Image Recognition via Convolutional Neural Network, *Electronics* 13 (2) (2024) 452.
- [30] S. Shakibhamedan, N. Amirafshar, A.S. Baroughi, H.S. Shahhoseini, N. Taherinejad, ACE-CNN: Approximate Carry Disregard Multipliers for Energy-Efficient CNN-Based Image Classification, *Regular Papers, IEEE Transactions on Circuits and Systems I*, 2024.
- [31] N. Yin, W. Pan, Y. Yu, C. Tang, Z. Yu, Low-Power Pass-Transistor Logic-Based Full Adder and 8-Bit Multiplier, *Electronics* 12 (15) (2023) 3209.
- [32] A. Salehi, M. Balasubramanian, DDCNet: Deep dilated convolutional neural network for dense prediction, *Neurocomputing* 523 (2023) 116–129.
- [33] M. Azizi, S. Talatahari, A.H. Gandomi, Fire Hawk Optimizer: A novel metaheuristic algorithm, *Artif. Intell. Rev.* 56 (1) (2023) 287–363.
- [34] T.H. Singh, P.T. Huang, K.S. Kao, C.S. Cheng, K.A. Wen, L.C. Wang, Energy-Efficient Sparse FFT and Compressed Transpose Memory for mmWave FMCW Radar Sensor System, *IEEE Trans. Instrum. Meas.* (2024).
- [35] S.K. Panda, K. Achyut, D.C. Panda, Synthesis and Time Analysis of FPGA-Based DIT-FFT Module for Efficient VLSI Signal Processing Applications, *Explainable Machine Learning Models and Architectures* (2023) 65–79.
- [36] L. Malathi, A. Bharathi, A.N. Jayanthi, FPGA design of FFT based intelligent accelerator with optimized Wallace tree multiplier for image super resolution and quality enhancement, *Biomed. Signal Process. Control* 88 (2024) 105599.
- [37] R. Guo, H. Liu, G. Xie, Y. Zhang, D. Liu, A self-interpretable soft sensor based on deep learning and multiple attention mechanism: From data selection to sensor modeling, *IEEE Trans. Ind. Inf.* 19 (5) (2022) 6859–6871.
- [38] R. Guo, Q. Chen, H. Liu, W. Wang, Adversarial Robustness Enhancement for Deep Learning-Based Soft Sensors: An Adversarial Training Strategy Using Historical Gradients and Domain Adaptation, *Sensors* 24 (12) (2024) 3909.