DESIGN AND VERIFICATION™
DVCON
CONFERENCE AND EXHIBITION
INDIA

September 10-11, 2025
Radisson Blu, Marathahalli, Bangalore

AI Accelerator for Autism Detection

Rompicharla Nokshit Rama Yaseswi
Bungatavala Mithun Chakravarthi
Yandarapu Jaya Kushal

BRISK-5687

Vellore Institute of Technology

Table of Contents

1. Table of Contents

List of Tables:

List Of Figures

Abstract

Autism spectrum disorder (ASD) is a neurodevelopmental disorder that needs early diagnosis for successful intervention. This Report discusses the design of a specialized AI accelerator for autism detection, with optimization of convolutional neural network (CNN) inference on edge-level hardware.

The system utilizes the Genesys 2 FPGA (Kintex-7) as the CNN accelerator and the Vega AT1051 as the host central processing unit. The FPGA processes computation-heavy operations like convolution, pooling, and dense layers, and the CPU (VEGA AT1051) manages image pre-processing and communication. Both sides of the transfer are supported using the AXI4 interface with GPIO supporting the control signal.

Hardware acceleration thus far outsizes a system implementing just CPU on its own to bring more-efficient and readily accessible AI-backed autism diagnosis. The project validates the possibility of running deep learning models on constrained devices with an aim to bring improved real-world diagnostic power in healthcare environments.

I. Introduction:

Autism is a condition that affects how people communicate, interact with others, and behave. It can look different in every person, which makes early diagnosis difficult. But finding autism early is important because it helps people get the right support at the right time. AI (Artificial Intelligence) is becoming a useful tool for this. Some AI models can look at facial features and small expressions in pictures to detect signs of autism. This is not a replacement for a doctor's diagnosis, but it can help by giving quick and consistent results, especially in places where autism specialists are not available.

The problem is that running AI models requires a lot of processing power, which many everyday devices don't have. Our project focuses on creating a special AI accelerator—a small, fast, and efficient chip—to help detect autism. This will allow AI models to work well even on simple, low-power devices. Our goal is to make autism detection easier and more affordable. By combining AI and embedded systems, we hope to create something useful that can help people. Even small improvements in this area could make a big difference.

II. Background Research

Recent advances in artificial intelligence (AI) and machine learning have helped improve early autism detection. AI models analyse different types of data, such as facial expressions, speech patterns, and brain scans, to identify autism spectrum disorder (ASD).

One effective method for autism detection is **eye-tracking technology**. Research shows that AI can study how a person's eyes move—tracking gaze patterns, quick eye movements, and attention shifts—to find differences between autistic and non-autistic individuals. Machine learning techniques, like Support Vector Machines and Deep Neural Networks, process this eye-tracking data to make accurate predictions. Another popular approach is using **deep learning models**, especially Convolutional Neural Networks (CNNs). These models analyse images, such as facial photos or heatmaps of eye movements, to detect patterns linked to autism. However, CNNs require a lot of computing power, which makes them difficult to run on everyday devices. This is why **hardware acceleration** is essential for real-time autism detection, allowing AI models to work efficiently on simpler, low-power devices.

CNN Accelerators for FPGA-Based AI Processing:

CNNs require a lot of calculations, especially multiply-accumulate (MAC) operations, which makes them difficult to run on small, low-power devices. To solve this problem, FPGA-based CNN accelerators use techniques like parallel processing, pipelining, and hardware optimizations to improve speed and reduce energy use.

One important method is Depth-wise Separable Convolution, which helps lower the number of calculations while keeping the model accurate. Instead of applying a complex filter to the entire image at once, this technique splits the process into two steps:

1. **Depth-wise Convolution** – Applies a filter to each colour channel (Red, Green, and Blue) separately.
2. **Pointwise Convolution** – Uses 1×1 filters to combine the results from different channels.

Using Depth-wise Separable Convolution on an FPGA reduces the amount of hardware needed and lowers power consumption, making CNNs more efficient for embedded systems.

Relevance to Our Project:

Given the computational intensity of CNN-based autism detection, implementing an FPGA-based CNN accelerator ensures low-latency inference and high energy efficiency. The Genesys 2 FPGA (Kintex-7 XC7K325T) provides sufficient resources, including 840 DSP slices for MAC operations, 16Mbit Block RAM (BRAM) for feature map storage, and high-speed AXI4 interfaces for communication with an external processor.

In our project, we will:

1. Use the VEGA processor to handle image preprocessing and AXI4 communication.
2. Implement a CNN accelerator on FPGA using depth wise separable convolution for efficient computation.
3. Leverage FPGA resources (DSPs, BRAM, and parallel processing) to improve inference speed.
4. This hybrid approach ensures real-time autism detection while optimizing resource utilization on the FPGA platform

III. Goal and Objectives:

The Goal:

Our goal is to design an efficient AI accelerator for autism detection, ensuring that advanced AI models can run smoothly on low-power devices for real-world healthcare applications.

Objectives:

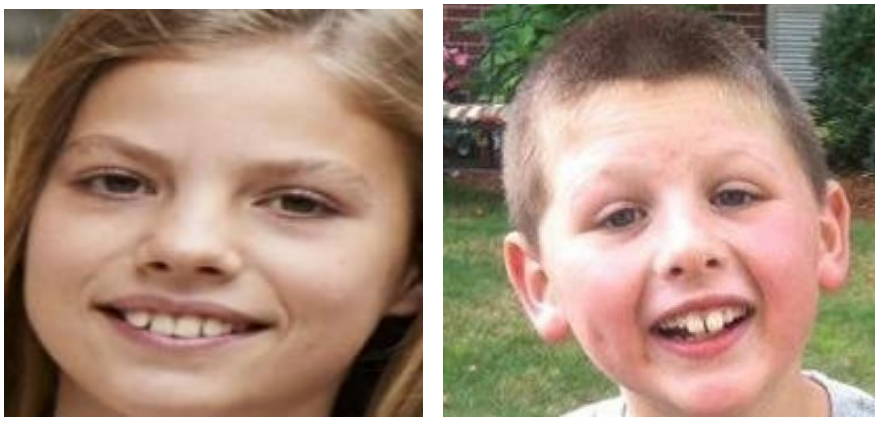
1. **Develop an Image-Based Autism Detection Model:** To Design a CNN-based model to detect early signs of autism from facial images, optimized for efficient hardware deployment.
2. **Design a Lightweight CNN Accelerator:** Create a hardware accelerator on the Genesys-2 FPGA that can handle key CNN operations like convolution and pooling. The aim is to offload these compute-heavy tasks from the main processor, improving speed and energy efficiency.
3. **Integrate with VEGA AT1051 Processor:** Establish a working interface between the VEGA processor and the accelerator using AXI4 and GPIO, ensuring smooth control communication and data exchange.
4. **Optimize for Real-Time Performance:** Aim to achieve inference within a 10–20 ms time window to meet the demands of real-time or near-real-time screening scenarios, without overloading the hardware.
5. **Prototype and Test on Actual Hardware:** Use the Genesys-2 board for prototyping, testing the complete pipeline under real conditions. Measure performance, resource utilization, and latency to assess practical feasibility.
6. **Promote Broader Access to AI in Healthcare:** Beyond the technical deliverables, this project seeks to highlight how edge-level AI solutions can help bridge gaps in early diagnosis, especially in underserved or remote communities where access to specialists may be limited.

IV. Design Process:

i. Problem Statement:

Detecting autism early is important for timely support. Traditional diagnosis depends on expert evaluation, which can be slow and hard to scale. AI-based facial analysis can help, but the challenge is to make it fast, efficient, and reliable in real-time. This project aims to build a system using a CNN accelerator on a Genesys 2 FPGA (Xilinx Kintex-7) with a Vega AT1051 RISC-V processor. The goal is to process images quickly while using less power. The system will securely transfer images, run CNN layers efficiently, and send results back to the processor.

Figure 1: Samples from Dataset:



Not Autistic Child **Autistic Child**

ii. Functional Specification:

1. Input Acquisition and Preprocessing

a) The Vega AT1051 processor captures the input image from an HDMI/USB camera and stores. It is in DDR3 memory (1GB) via its DDR3 controller.

b) The preprocessing steps include:
- i) Resizing the image to 256×256 resolution.
- ii) Normalization to scale pixel values appropriately.
- iii) Fixed-point conversion for optimized FPGA computation.

c) The pre-processed image is prepared for transfer to the FPGA CNN accelerator.

2. Data Transfer and Memory Management

a) CNN model weights and biases are stored in FPGA BRAM for fast access.
b) The AXI4 interconnect is used for communication between Vega and FPGA:
- i) AXI4-Stream for high-speed image/feature map transfer.
- ii) AXI4-Lite for configuration and control signals.

c) Pre-processed image data is transferred from DDR3 to FPGA CNN accelerator via AXI4.

3. CNN Inference on FPGA

a) The CNN accelerator (Implemented in C and synthesized using Vitis HLS) performs neural network computations.
b) The FPGA efficiently utilizes its resources:
- i) BRAM for storing weights and feature maps.
- ii) DSP slices for convolution, activation, and matrix multiplication.

c) The CNN accelerator executes:
- i) Convolution layers.
- ii) Pooling layers.
- iii) Batch Normalization.
- iv) Fully Connected layers for final classification.

d) The classification result is stored back in DDR3 memory.

4. Communication and Output Processing

a) Vega fetches the classification result via AXI4.
b) The decision logic is executed in Vega to determine autism presence.
c) The final output is:
    i) Displayed using FPGA GPIO LEDs.
    ii) Optionally sent to an external display via VGA/HDMI.
d) Vega also manages interrupts, processing status, and debug information. The Entire process is written and implemented in C using Vitis Embedded(SDK).

iii. Proposed Design:

System Architecture: The system is composed of two major subsystems:

a) **Vega AT1051 (Host CPU):**
   b. **Functions:** Performs preprocessing, transfers weights/images, configures FPGA through AXI4-Lite, and handles GPIO-based interrupts.
   c. **Interfaces:** AXI4 for data, AXI4-Lite for control, GPIO for interrupts.

b) **Genesys 2 FPGA (Kintex-7):**
   a. **Functions:** Executes CNN inference using an internal accelerator.
   b. **Resources:**
      i. **DSP slices:** High-speed convolution and dense layer operations.
      ii. **BRAM (16 Mbits/2MB):** Stores images, weights, and intermediate data.
      iii. **LUTs/Flip-Flops:** Control logic and sequencing.
      iv. **DDR3 (1 GB):** External memory for data buffering via DDR3 controller.
      v. **Interconnect:** AXI4-32bit for internal communication.

Figure 2: Proposed System Architecture for Autism Detection on FPGA

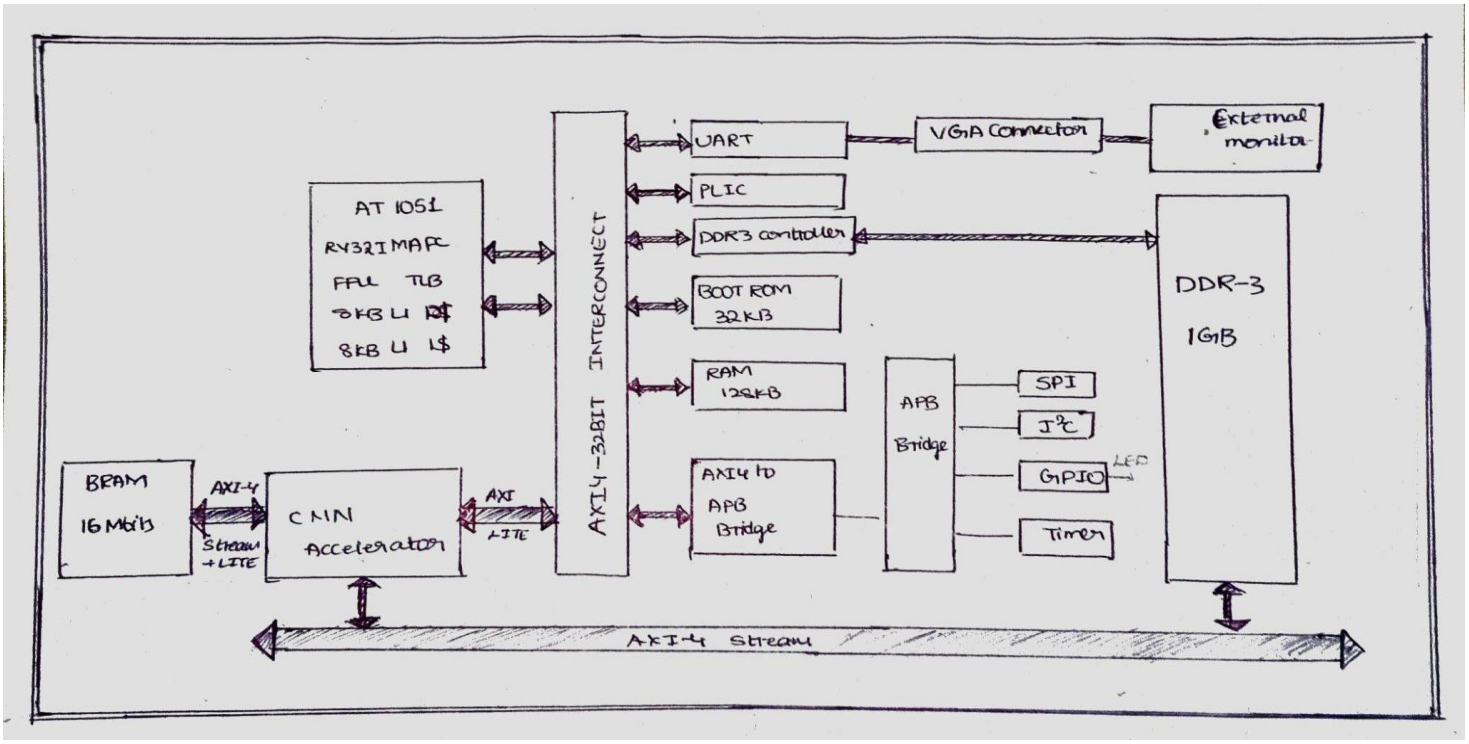


Table 1-Task Distribution

Task	**Executed By**	**Rationale**
Preprocess and convert image	Vega AT1051 CPU	Efficient software execution
Load CNN weights and image	Vega → FPGA (AXI4)	Host controls memory/data management
Execute CNN inference	Genesys 2 FPGA	Parallelism reduces computation time
Communicate result	FPGA → Vega (AXI4)	Low-latency response mechanism
Display and further processing	Vega AT1051 CPU	Manages user interface and logic

iv. Analysis of Final Design:

a. **Inference Speed:**
The FPGA-based accelerator is expected to achieve 10–20ms inference latency due to parallel processing with DSP slices, compared to 100–500ms on CPU.

b. **Power Efficiency:**
FPGA computation is expected to consume less power than equivalent CPU execution.

c. Resource Utilization:

i. **DSP Slices:** Utilized for convolution and dense operations

ii. **BRAM:** Stores weights, image, and intermediate feature maps.

iii. **LUTs/Flip-Flops:** Used for control FSM and internal sequencing.

d. **Scalability and Flexibility:**
Modular design enables layer expansion and integration of additional AI models. AXI4-based architecture accommodates scaling in data bandwidth and type.

e. **Practical Considerations:**
The system architecture reflects a realistic embedded scenario. FPGA and host CPU coordination is achieved using AXI4 and GPIO interfaces, known for reliability in industrial applications.

v. Testing and Implementation of Final Design:

1. **Simulation and Verification:**
   a. Validate CNN modules (Convolution, Pooling, Batch Norm) using C testbenches in Vitis HLS.
   b. Test AXI4-Stream & AXI4-Lite interfaces for data transfer between Vega and FPGA.
   c. Ensure DDR3 and BRAM controllers function correctly for image and weight storage.

2. **FPGA Implementation & Debugging:**
   a. Synthesize the design in Xilinx Vivado and check timing/resource utilization.
   b. Debug using ILA (Integrated Logic Analyzer) for on-chip signal monitoring.

3. **Hardware Testing on Genesys 2 Board:**
   a. Input Flow: Capture image via HDMI/USB camera, store in DDR3, and send to FPGA.
   b. CNN Processing: Verify FPGA classification output against software results.

c. End-to-End Testing: Ensure FPGA signals inference completion via GPIO to Vega.

4. **Performance Analysis & Optimization:**
   a. Measure latency, AXI4 bus efficiency, and FPGA power consumption.
   b. Optimize BRAM/DSP slice usage for better resource efficiency.

5. **Deployment & Final Integration:**
   a. Load the bitstream onto Genesys 2 FPGA.
   b. Perform final real-world testing for autism detection accuracy.

V. Results and Discussion

Our trained model achieved an **accuracy of 88.5%** using a dataset consisting of **5280 samples** indicating promising potential of the model for early autism detection. These results are encouraging, but we think this model needs to be trained on larger and more diverse datasets for more promising results before deployment into real-world clinical settings. We believe this is necessary due to the inherent sensitivity in healthcare, where even a 1% increase in used instrument's accuracy can significantly impact patient outcomes and save many lives.

The conversion of our CNN model from floating-point to fixed-point representation on the FPGA is expected to result in classification accuracy that closely mirrors the original software implementation, with only minimal degradation that can be refined through further optimization. These minor quantization errors may affect the detection of subtle features. This issue can be mitigated through quantization-aware training and iterative calibration on larger, more diverse datasets, ensuring that the model remains robust even in edge cases.

On the hardware front, although the current FPGA platform can demonstrate efficient performance, there is potential for further enhancements. Upgrading the number of DSP slices available for use and expanded on-chip memory (BRAM) could allow for the integration of more complex models or simultaneous processing pipelines, ultimately reducing latency even further. Additionally, exploring hybrid accelerator architectures—such as combining FPGA with an embedded GPU or ASIC—may improve power efficiency and speed. Incorporating dynamic reconfiguration and advanced error-correction techniques could also help minimize fixed-point computation discrepancies, ensuring more consistent and reliable outputs in clinical settings.

VI. Conclusion

Our project represents an important step toward early autism detection by combining advanced AI with FPGA-based hardware acceleration. While our initial results are promising, we recognize that further refinement is needed. As we participate in this design contest, our focus is on refining this model and exploring innovative ideas for future enhancements, always with a realistic understanding of the challenges involved.

We acknowledge that autism is a complex condition and that our system is not intended to completely replace professional diagnosis. Instead, we intend for it to serve as a supportive tool, complementing traditional clinical methods with a technology-driven perspective. Our work to date and our ongoing efforts underscore the transformative potential of AI in healthcare, and we remain dedicated to continuously refining our design based on expert insights and evolving real-world challenges.

VII. References

1. https://www.physio-pedia.com/Autism_Spectrum_Disorder_and_Motor_Development
2. ACM SIGCHI. (2024, May 8). *Early Autism Screening in Children Using Facial Recognition*. YouTube. https://www.youtube.com/watch?v=iXoCNB9a5uI
3. Bai, L., Zhao, Y., & Huang, X. (2018). A CNN Accelerator on FPGA Using Depth wise Separable Convolution. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *65*(10), 1415–1419. https://doi.org/10.1109/tcsii.2018.2865896
4. Kollias, K.-F., Syriopoulou-Delli, C. K., Panagiotis Sarigiannidis, & Fragulis, G. F. (2022). *Autism detection in High-Functioning Adults with the application of Eye-Tracking technology and Machine Learning*. 1–4. https://doi.org/10.1109/mocast54814.2022.9837653
5. Pushpmala Nawghare, & Prasad, J. R. (2024). *Early Detection of Autism Spectrum Disorder Using AI and Machine Learning Models: A Systematic Review for Effective Intervention*. 1–6. https://doi.org/10.1109/punecon63413.2024.10895404
6. AutismScienceFd. (2011, March 18). *Dr. Kasia Chawarska uses face scanning and recognition to perceive social abnormalities in autism*. YouTube. https://www.youtube.com/watch?v=YgNPW1jqguw
7. Srivastava, H., & Kishor Sarawadekar. (2020). *A Depthwise Separable Convolution Architecture for CNN Accelerator*. https://doi.org/10.1109/aspcon49795.2020.9276672
8. https://github.com/HemaKumar0077/Autism-Detection-by-image/blob/main/Autism_original.ipynb (Source Code)