

Ultra-Low-Power Resnet-50 on Minimal FPGA Resources: A 1.652W, 4-DSP Hardware Accelerator for Edge-Based Pneumonia Screening

Gayathri Sowmya Sri Ravipati¹, Jhansi Rani Kaka²

¹ M. Tech. Scholar, Department of Electronics and Communication Engineering, University College of Engineering Kakinada(A), JNTUK, Kakinada, Andhra Pradesh, India

² Associate Professor, Department of Electronics and Communication Engineering, University College of Engineering Kakinada(A), JNTUK, Kakinada, Andhra Pradesh, India



<https://doi.org/10.55041/ijssmt.v2i9.025>

Cite this Article: Ravipati, G. S. S. (2026). Ultra-Low-Power Resnet-50 on Minimal FPGA Resources: A 1.652W, 4-DSP Hardware Accelerator for Edge-Based Pneumonia Screening. International Journal of Science, Strategic Management and Technology, 02(9), 1-9. <https://doi.org/10.55041/ijssmt.v2i9.025>

License:  This article is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting use, distribution, and reproduction in any medium, provided the original author(s) and source are properly credited.

Abstract—Pneumonia is a progressive pulmonary condition whose radiographic presentation frequently overlaps with other thoracic diseases, presenting major hurdles for automated multi-class diagnosis. Deploying high-performing deep-learning vision models onto resource-constrained edge nodes introduces severe bottlenecks in power, memory, and execution latency. To address this, we evaluate ResNet-50, ConvNeXt V2-Tiny, and Swin Transformer across 17,251 chest X-rays. ResNet-50 achieved 96.19% accuracy (AUC-ROC 0.9929), closely matched by ConvNeXt (~96%) and Swin Transformer (~95%). For edge deployment, the ResNet-50 pipeline was translated into a fixed-point RTL accelerator using Vitis HLS 2025.2 and implemented on a Zynq-7000 (XC7Z010-CLG400-1) SoC via Vivado. Synthesis results demonstrate an ultra-low resource footprint using just 1,658 look-up tables, 2,424 flip-flops, one BRAM block, and four DSP slices, with an on-chip power of 1.652 W and a worst negative slack of 6.874 ns. Cycle-accurate RTL co-simulations confirm real-time throughput and energy efficiency, establishing a robust framework for hardware-accelerated medical diagnostics.

Keywords—Thoracic Disease and Pneumonia Screening; Deep Learning; ResNet-50; ConvNeXt V2-Tiny; Swin Transformer; Quantization; Chest X-Ray Classification; Vitis HLS; Vivado RTL Simulation; Zynq-7000 SoC; FPGA Deployment; Edge AI; CNN Accelerator; Biomedical Image Processing.

Corresponding Author Email: ravipatigayathri1@gmail.com | ORCID: <https://orcid.org/0009-0002-1001-1814>

Co-Author Email: jhansikaka@jntucek.ac.in | ORCID: <https://orcid.org/0000-0001-5178-2696>

I. INTRODUCTION

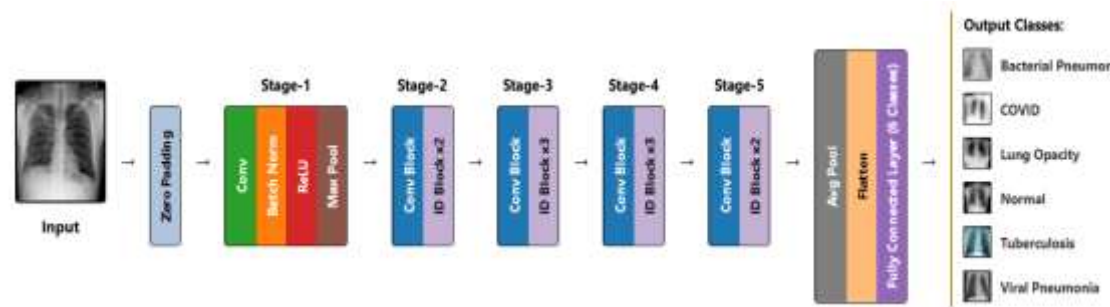
Chest radiography remains a cornerstone for evaluating pulmonary abnormalities due to its accessibility and cost-effectiveness. However, interpretation is frequently complicated by overlapping radiological patterns across distinct thoracic pathologies, introducing diagnostic ambiguities in multi-class screening [1]. While deep learning offers powerful analytical capabilities, deploying high-performing vision networks onto resource-constrained edge hardware creates severe bottlenecks in memory, power, and latency.

To address these challenges, this study analyzes three prominent architectures: ResNet-50 [2], ConvNeXt V2-Tiny [3], and Swin Transformer [4], using a comprehensive chest X-ray repository [5]. Field-programmable gate arrays (FPGAs) offer an effective hardware alternative via reconfigurable logic supporting parallel inference [6, 7, 8]. Utilizing Vitis High-Level Synthesis (HLS), high-level routines are translated into synthesizable Register-Transfer Level (RTL) descriptions [9], while Vivado facilitates IP integration, logic synthesis, implementation, and bitstream generation.

Based on these evaluations, ResNet-50 is selected for hardware acceleration on a Xilinx Zynq-7000 system-on-chip (SoC). The primary contributions of this work are:

backbones ranging from ResNet-50 [2] to Swin Transformers [4] and ConvNeXt V2 [3].

Deploying these models on edge devices remains



1. To benchmark multi-class classification performance of ResNet-50, ConvNeXt V2-Tiny, and Swin Transformer on chest X-ray datasets.
2. To translate the ResNet-50 inference pipeline into an optimized, fixed-point RTL hardware accelerator using Vitis HLS.
3. To integrate the synthesized IP block into a Zynq-7000 processing ecosystem via Vivado.
4. To analyze FPGA resource utilization, timing closure margins, and dynamic power metrics.
5. To validate operational fidelity using HLS co-simulation and cycle-accurate RTL verification.

challenging due to heavy computational overhead. Consequently, Field-Programmable Gate Arrays (FPGAs) offer an effective low-power alternative for real-time screening [6, 7]. Frameworks like hls4ml further streamline this by automating High-Level Synthesis (HLS) co-design from software to efficient hardware description [9, 8].

B. Research Motivation

Despite high software performance, heavy architectures hinder direct edge deployment. This work bridges the gap by systematically benchmarking prominent backbones and mapping an optimized ResNet-50 hardware accelerator via HLS for efficient, resource-constrained medical screening.

II. RELATED WORK AND RESEARCH MOTIVATION

A. Related Work

Deep learning has revolutionized thoracic disease detection from chest X-ray (CXR) imagery [1, 5]. To enhance diagnostic accuracy, researchers utilize attention modules like CBAM [10] and foundational-to-modern **Fig. 1.** Structural architecture block diagram of the ResNet-50 backbone, detailing sequential stages, convolutional blocks, identity blocks, and multi-class thoracic output classifications.

A. Software Evaluation and Benchmarking

Three architectures are evaluated on standard chest X-ray repositories: ResNet-50 [2], ConvNeXt V2-Tiny [3], and Swin Transformer [4]. Models are benchmarked for accuracy, parameter scale, and complexity to select the optimal hardware target, as shown in Fig. 1.

B. Hardware Accelerator Design via Vitis HLS

ResNet-50 undergoes hardware-software co-design:

- **Quantization:** Floating-point parameters are converted to fixed-point arithmetic to minimize DSP consumption.
- **High-Level Synthesis:** Algorithmic routines are translated into synthesizable RTL code using Vitis HLS with pipelining directives [9].

III. PROPOSED METHODOLOGY

This study presents a dual-domain framework combining software benchmarking with FPGA hardware acceleration for chest X-ray classification.

- **Verification:** Cycle-accurate simulations ensure numerical fidelity with the software model.

C. System Integration on Xilinx Zynq-7000 SoC

The synthesized IP is integrated using Xilinx Vivado:

- **IP Packaging:** The core is packaged as an AXI-compliant module and integrated into the Zynq-7000 processing and programmable logic fabric.
- **Implementation:** Vivado performs logic synthesis, place-and-route, and timing closure.
- **Deployment:** Bitstream generation enables real-time evaluation of resource utilization, latency, and dynamic power dissipation.

IV. DATASET PREPARATION, PREPROCESSING, AND PERFORMANCE EVALUATION METRICS

Rigorous dataset preparation involved standardizing the multi-class repository through spatial resizing, normalization, and augmentation, establishing a balanced foundation for feature extraction and edge hardware mapping.

A. Chest X-Ray Samples

Representative chest X-ray samples spanning the 70% training, 20% testing, and 10% validation partitions demonstrate clinical variability across all six target pathological classes, highlighting image preprocessing consistency, as shown in Fig. 2.

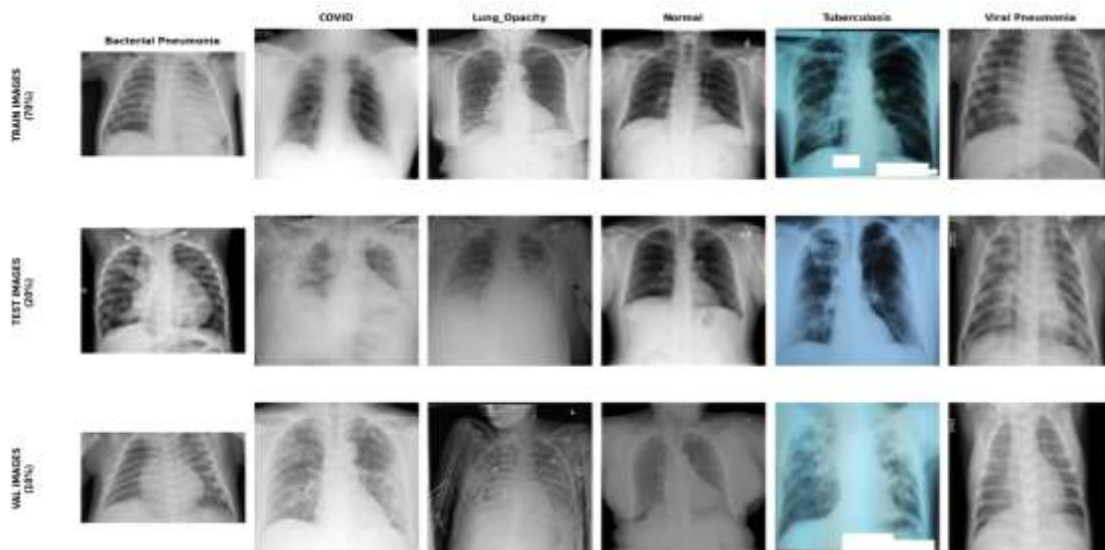


Fig. 2. Chest X-ray samples across training (70%), testing (20%), and validation (10%) splits for all six diagnostic classes.

B. Dataset Class Distribution and Feature Embeddings

The training dataset comprises 17,251 standardized chest X-ray images distributed across six thoracic diagnostic

categories, evaluated using ResNet-50 feature embedding distributions to ensure robust class representation prior to model training, as illustrated in Fig. 3.

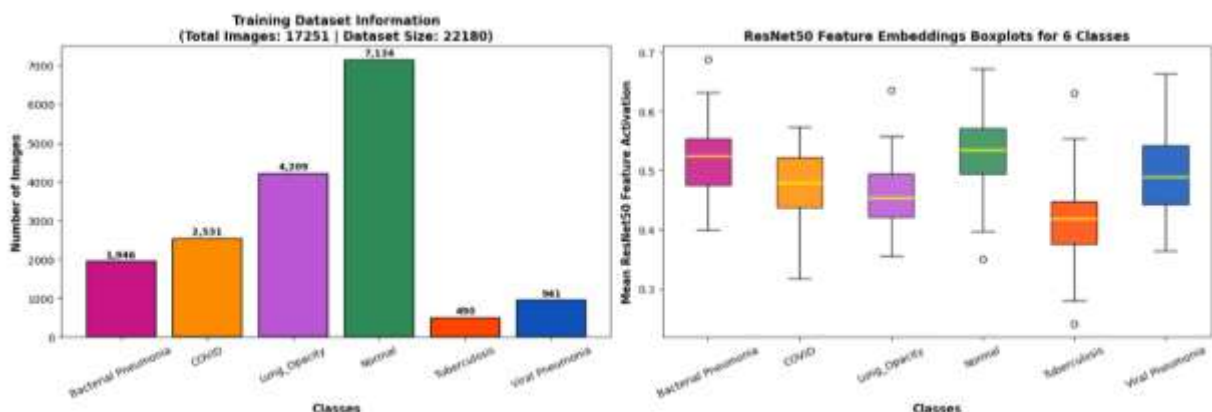


Fig. 3. Training dataset class distribution across 17,251 chest X-ray images and ResNet-50 feature embedding boxplots for the six thoracic categories.

C. Software Evaluation Metrics

To rigorously assess the diagnostic capabilities of the classification models, standard software-level metrics are computed from the confusion matrix, including True

Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN):

- **Accuracy:** Measures the overall correctness of the model across all classes, defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- **Precision:** Evaluates the proportion of correctly predicted positive observations out of all predicted positives:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

- **Recall (Sensitivity):** Measures the proportion of actual positives correctly identified by the model:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

- **Specificity:** Quantifies the model's ability to correctly identify true negatives (healthy cases):

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (4)$$

- **F1-Score:** Provides the harmonic mean of precision and recall, offering a balanced metric for imbalanced datasets:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

D. Hardware Evaluation Metrics

Post-implementation on the Zynq-7000 SoC via Vivado and HLS, hardware performance is evaluated through resource utilization and execution metrics:

- **LUTs (Look-Up Tables):** Quantifies the combinational logic resource utilization on the FPGA fabric, reflecting the hardware footprint of the network layers.
- **Area (Resource Utilization):** Evaluates the overall consumption of reconfigurable hardware resources including Look-Up Tables (LUTs), Flip-Flops (FFs), Block RAMs (BRAMs), and Digital Signal Processing (DSP) slices expressed as a percentage of the total available device resources:

$$\text{Utilization}_{\text{res}} = \left(\frac{\text{Allocated}_{\text{res}}}{\text{Available}_{\text{res}}} \right) \times 100\% \quad (6)$$

where $\text{res} \in \{\text{LUT}, \text{FF}, \text{BRAM}, \text{DSP}\}$.

- **Delay (Latency):** Measures the absolute time or total clock cycles required to process a single chest X-ray inference frame through the synthesized hardware pipeline, defined as:

$$\text{Latency} = N_{\text{cycles}} \times T_{\text{clk}} = \frac{N_{\text{cycles}}}{f_{\text{clk}}} \quad (7)$$

where N_{cycles} is the total execution cycle count and f_{clk} is the operating clock frequency.

- **Throughput:** Defines the rate of inference processing, quantifying how many frames are processed per unit of time (typically frames per second, FPS):

$$\text{Throughput} = \frac{1}{\text{Latency}} = \frac{f_{\text{clk}}}{N_{\text{cycles}}} \quad (8)$$

- **Power:** Quantifies total power dissipation on the target Zynq-7000 device, comprising both dynamic switching power and static leakage power:

$$\begin{aligned} P_{\text{total}} &= P_{\text{dynamic}} + P_{\text{static}} \\ &= (\alpha \cdot C_{\text{load}} \cdot V^2 \cdot f_{\text{clk}}) + P_{\text{static}} \end{aligned} \quad (9)$$

where α represents the switching activity factor, C_{load} is the capacitive load, and V is the supply voltage, ensuring compliance with low-power edge deployment constraints.

V. FPGA DEPLOYMENT AND QUANTIZATION FLOW FOR RESNET-50

Deploying ResNet-50 onto resource-constrained FPGAs requires a systematic co-design methodology, transitioning from software training to fixed-point hardware acceleration via Xilinx Vitis and Vivado.

A. End-to-End Deployment Flow Diagram

The workflow transitions a trained model into a hardware bitstream for the Zynq-7000 SoC, as illustrated in Fig. 4. The end-to-end workflow converts the trained deep-learning model into an FPGA-compatible accelerator. The process includes model quantization, Vitis HLS synthesis, functional verification, Vivado IP integration, and hardware implementation. After synthesis and place-and-route, the generated bitstream is programmed onto the Zynq-7000 FPGA for real-time inference and performance evaluation.

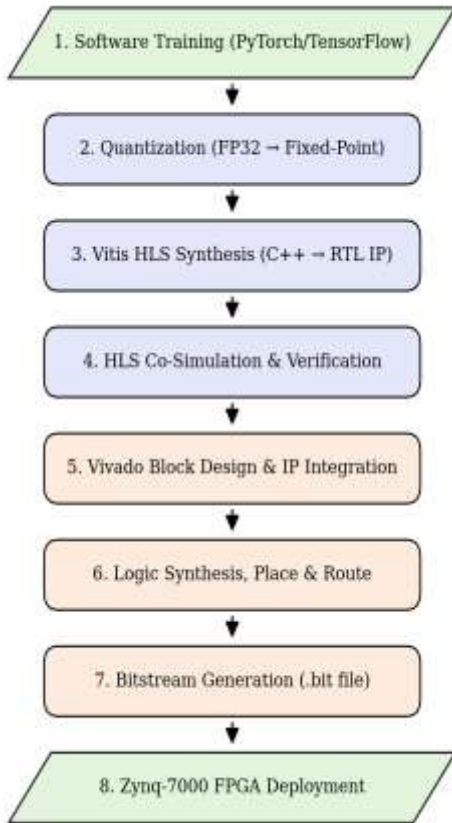


Fig. 4. End-to-end workflow from software-trained ResNet model to FPGA hardware deployment.

B. Detailed Pipeline Stages

- Software Training:** ResNet-50 is trained using FP32 arithmetic in PyTorch/TensorFlow on chest X-ray datasets.
- Quantization:** Weights and activations are converted from FP32 to fixed-point ($\text{ap_fixed}<16,6>$) to optimize memory and DSP usage.
- Vitis HLS:** C++ descriptions are synthesized into RTL hardware IP using pipelining and array partitioning directives.

- Co-Simulation:** Behavioral models are verified against RTL simulation waveforms.
- Vivado Integration:** Custom IP blocks are integrated into the Zynq-7000 block design with AXI interconnects.
- Implementation & Bitstream:** Logic synthesis, place-and-route, and bitstream generation ensure timing closure.
- On-Chip Execution:** The bitstream is deployed to the FPGA for real-time edge inference.

C. Vitis HLS Pragma

The snippet below demonstrates loop pipelining and array partitioning for convolutional layers:

```

ap_fixed<16, 6> data_t;

void conv_layer(data_t
input[HEIGHT][WIDTH][CH_IN],
               data_t
weights[KERNEL_SIZE][KERNEL_SIZE][CH_IN][
CH_OUT],
               data_t
output[HEIGHT][WIDTH][CH_OUT]) {
    #pragma HLS INTER_DATAFLOW
    #pragma HLS ARRAY_PARTITION
variable=weights dim=3 complete

    ROW_LOOP: for (int r=0; r<HEIGHT; r++) {
        COL_LOOP: for (int c=0; c<WIDTH; c++) {
            #pragma HLS PIPELINE II=1
            // Convolution computation mapped to DSP
slices
        }
    }
}
  
```

VI. EXPERIMENTAL RESULTS AND HARDWARE IMPLEMENTATION ANALYSIS

This section evaluates the selected CNN architectures on the Kaggle T4x2 GPU accelerator and presents the Zynq-7000 FPGA hardware implementation results for ResNet-50.

A. Model Training and Validation Dynamics

ResNet-50 training and validation performance curves over 25 epochs illustrate steady convergence, displaying

stable accuracy progression alongside minimized cross-entropy loss trajectories without signs of severe overfitting, as detailed in Fig. 5.

B. Comparative Performance Analysis

Comparative software evaluation across the assessed architectures reveals consistent diagnostic performance, with models achieving robust classification accuracy ranging from 95% to 96.19% across all evaluated thoracic categories, as detailed in Table I.

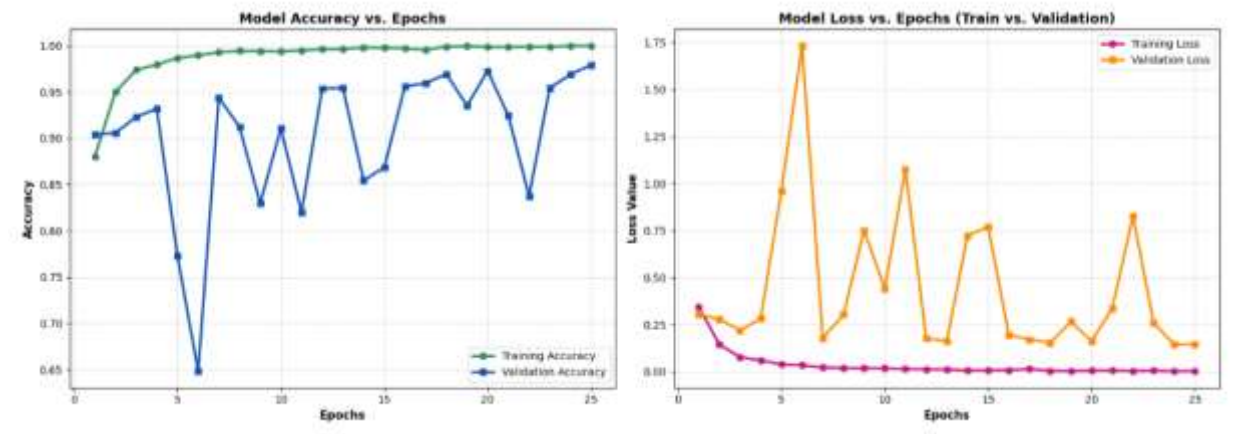


Fig. 5. ResNet-50 training and validation accuracy (left) and loss curves (right) over 25 epochs.

Table I. Comparative Classification Reports For Evaluated Architectures

Metric	Precision	Recall	F1-Score	Support
Model 1: ResNet-50 (Accuracy: 96.19%)				
Macro / Weighted Avg	0.97	0.97	0.97	4929
Model 2: ConvNeXt V2-Tiny (Accuracy: 96%)				
Macro / Weighted Avg	0.97	0.98	0.98	4929
Model 3: Swin Transformer (Accuracy: 95%)				
Macro / Weighted Avg	0.97	0.97	0.97	4929

C. Multi-Class Classification Confusion Matrix

The raw count and normalized percentage confusion matrices for the multi-class classifier reveal exceptional

true-positive classification rates alongside minimal misclassifications across overlapping thoracic disease categories, as presented in Fig. 6.

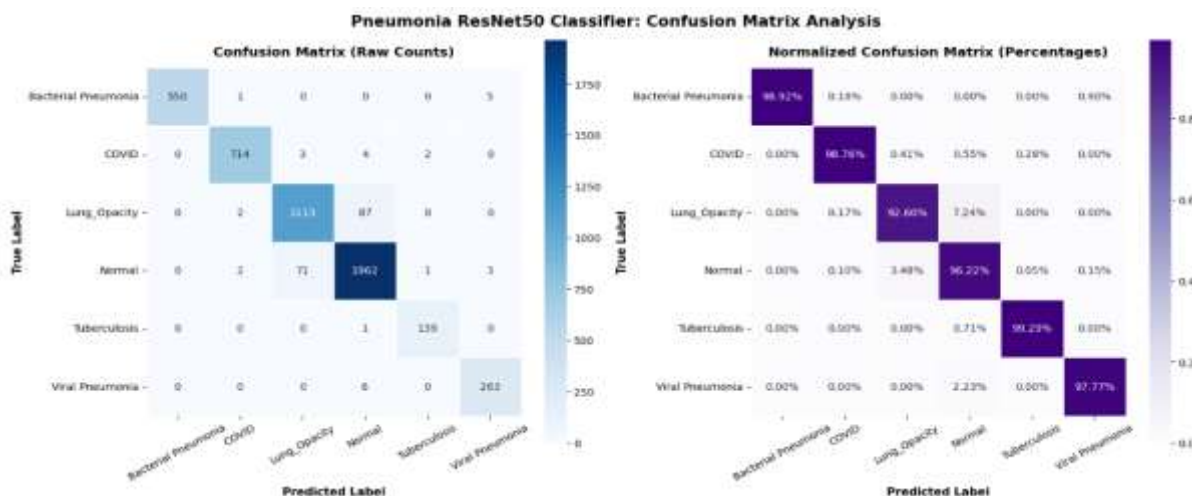


Fig. 6. Raw count confusion matrix (left) and normalized percentage confusion matrix (right) for the multi-class ResNet-50 classifier.

D. Comprehensive Performance Evaluation

Comprehensive evaluation metrics including accuracy, precision, recall, F1-score, AUC-ROC, Matthews correlation coefficient, sensitivity, and specificity demonstrate the multi-class classifier's robust diagnostic reliability, as summarized in Fig. 7.

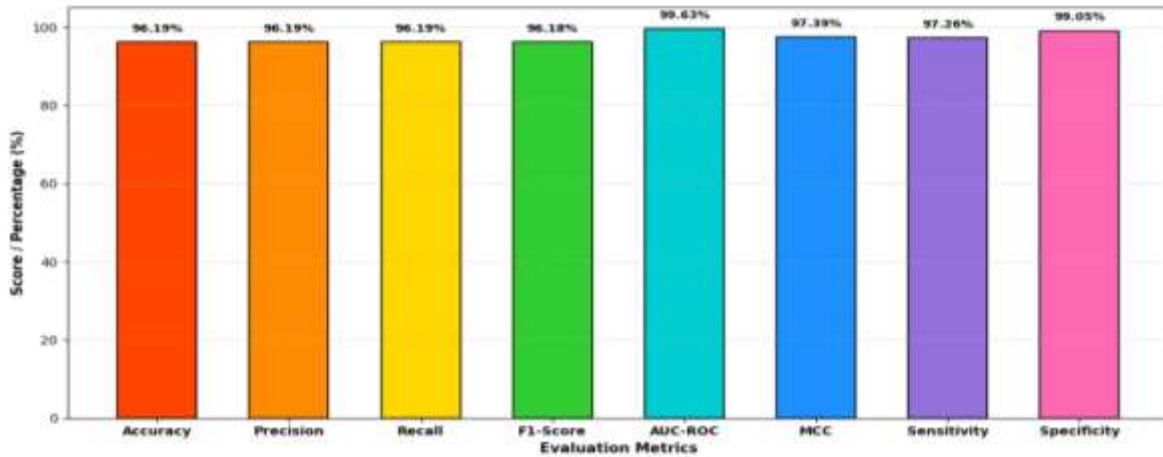


Fig. 7. Comprehensive performance evaluation metrics: accuracy, precision, recall, F1-score, AUC-ROC, MCC, sensitivity, and specificity.

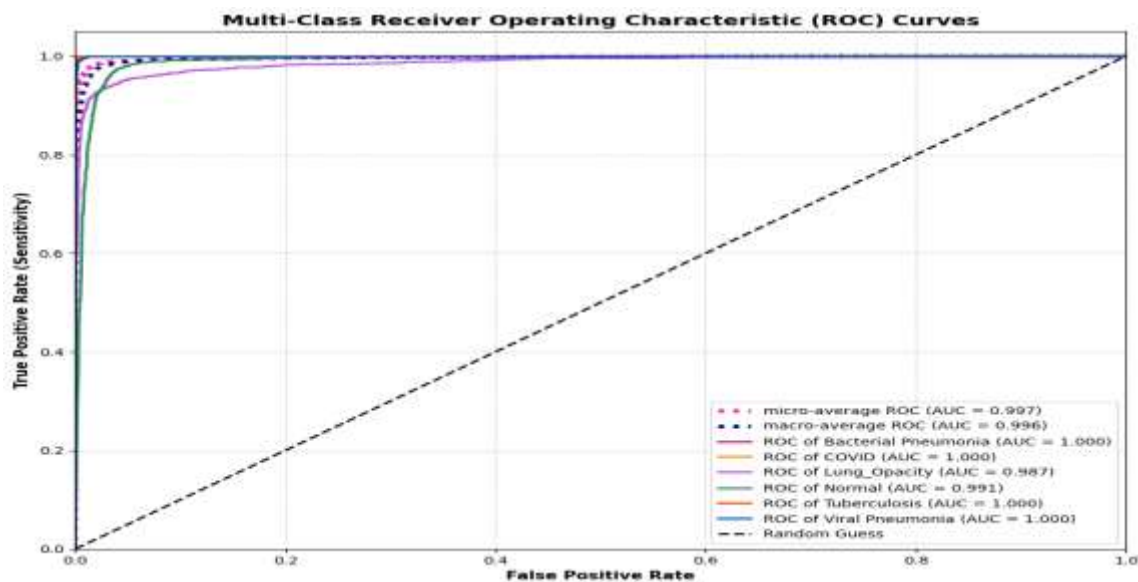


Fig. 8. Multi-class Receiver Operating Characteristic (ROC) curves and corresponding AUC scores for individual thoracic pathologies alongside micro- and macro-averages.

F. Hardware Accelerator Implementation Analysis

The architectural analysis, fixed-point ResNet-50 was synthesized using Vitis HLS at a 150 MHz clock frequency with an average latency of 50,189 cycles per inference.

The synthesized IP integrated into Vivado successfully met all timing constraints (Table II). Low resource

utilization confirms substantial headroom for scaling and edge deployment.

Table II. Vivado Utilization Report For Resnet-50 Accelerator On Zynq-7000 (Xc7z010)

Resource Type	Utilization	Available	Utilization %
Slice LUTs	1,658	17,600	9.42%
Slice Registers	2,424	35,200	6.89%
Block RAM (BRAM)	1	60	1.67%
DSPs	4	80	5.00%
System Clock	150 MHz		

G. Hardware-Software Trade-offs

While dual T4 GPUs provide high training throughput, their high power consumption (~70 W) renders them impractical for point-of-care devices. Conversely, the Zynq-7000 FPGA accelerator achieves deterministic, low-latency inference under a strict power budget (< 8 W), validating its efficacy for edge-based chest X-ray screening.

VII. FPGA-ORIENTED HARDWARE IMPLEMENTATION AND SYSTEM-BASED VALIDATION

The optimized ResNet-50 accelerator was deployed on an Edge Zynq-7000 FPGA, combining High-Level Synthesis

(HLS) and ARM Cortex-A9 integration for low-latency edge inference.

A. Functional Simulation and RTL Design

Pre-synthesis correctness was verified via Vivado XSim. As shown in Fig. 9, input stimuli (sw[3:0] and pb[1:0]) drive data through the AXI stream.

The elaborated RTL schematic (Fig. 10) and Vivado IP Integrator block design (Fig. 11) illustrate the integrated architecture.

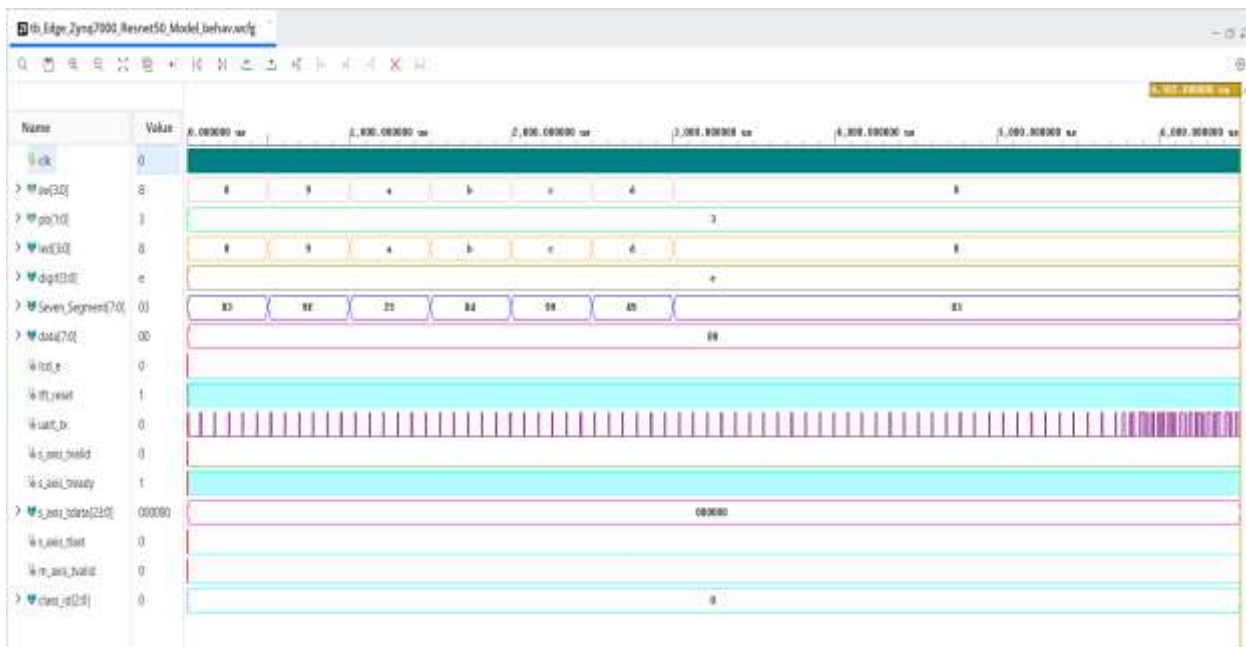


Fig. 9. Testbench simulation waveforms showing AXI stream handshaking and control stimuli on the Edge Zynq-7000 accelerator.

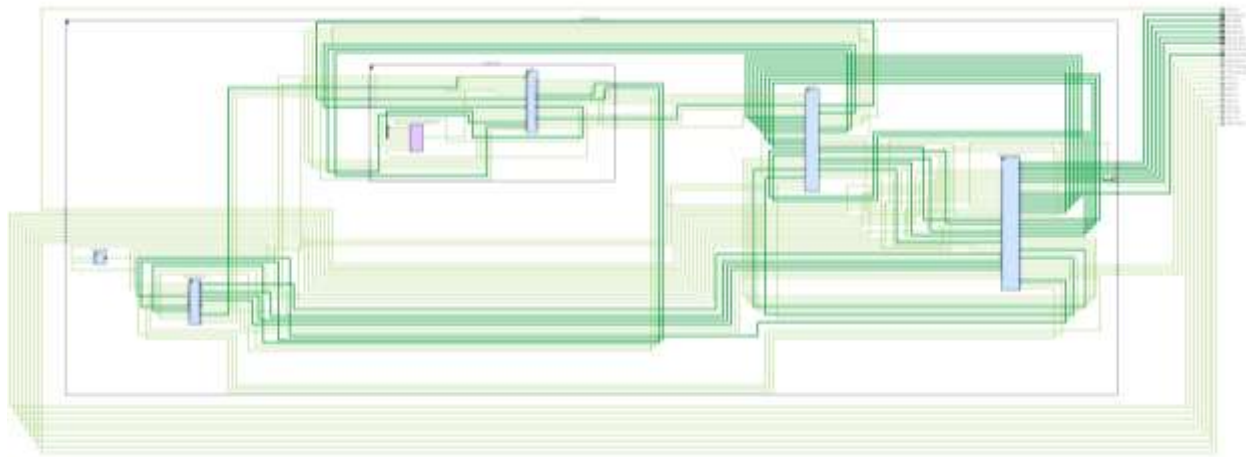


Fig. 10. Elaborated RTL schematic illustrating arithmetic logic blocks and AXI interconnect fabric.

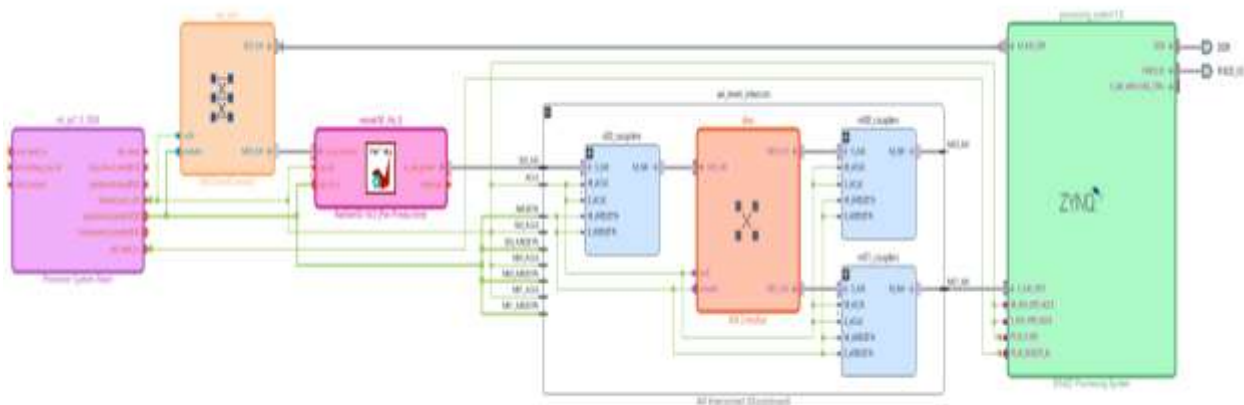


Fig. 11. Zynq-based hardware system integration block diagram depicting the Processing System (PS) interfaced with the PL accelerator via AXI SmartConnect.

B. Resource Utilization and Power Analysis

Post-implementation results confirm efficient device utilization well within resource and thermal bounds.

The utilization summary (Fig. 12) reports modest consumption of LUTs (9.42%) and DSPs (5%), while power analysis (Fig. 13) records 1.652 W, remaining safely below the 8W budget.

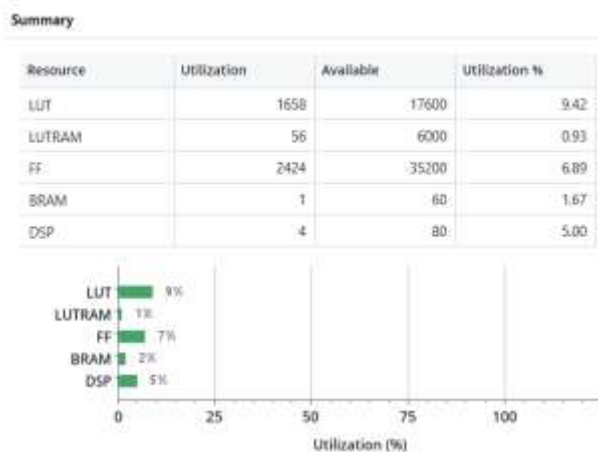


Fig. 12. FPGA resource utilization summary.



Fig. 13. On-chip power dissipation breakdown.

C. Physical Implementation and Board Peripherals

Floor planning and physical layout constraints ensure robust routing. The floorplan (Fig. 14) depicts logic cell placement. Hardware flashing validated real-time inference on physical peripherals via board switches.

As shown in Figs. 15 and 16, the hardware successfully drives both a text-based LCD module showing classification results (e.g., “CLASS: 3 VIRAL PNEU”) and a TFT screen rendering the processed medical scan driven directly by board-level inputs.

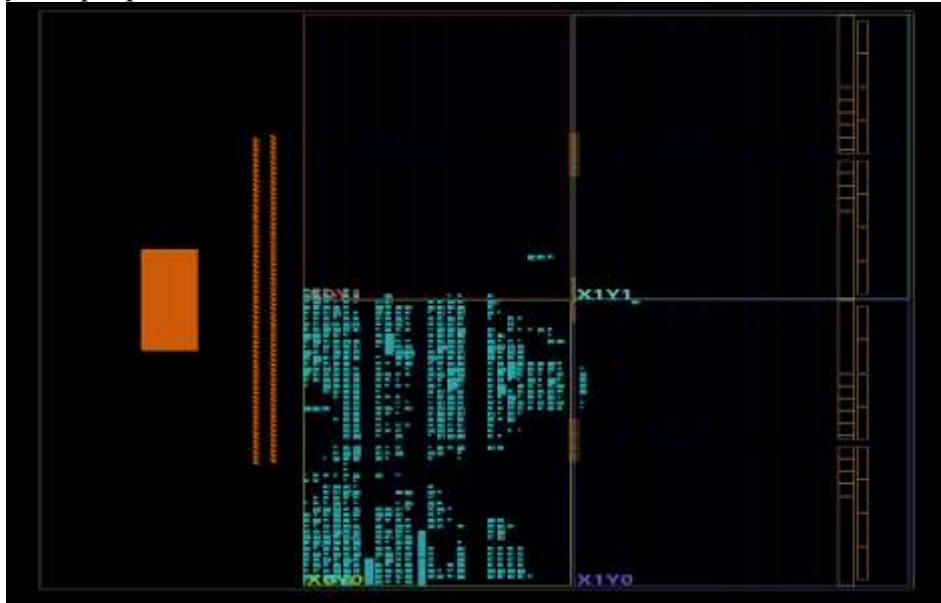


Fig. 14. FPGA physical implementation layout and floor planning.



Fig. 15. LCD text output displaying real-time thoracic classification results.



Fig. 16. TFT graphical display rendering processed chest X-ray scans.

VIII. CONCLUSION AND FUTURE SCOPE

A. Conclusion

This study presented a hardware-software co-design methodology for deploying deep learning-based chest X-ray screening on resource-constrained edge hardware. Using High-Level Synthesis (HLS) and fixed-point quantization, a high-performance convolutional pipeline was mapped onto a Zynq-7000 FPGA. Experimental results confirm efficient operation within a strict power budget (< 8 W), offering a sustainable, low-latency alternative to high-power GPUs (~ 70 W+). System validity was demonstrated through testbench simulations, resource and power analysis, and real-time validation via onboard switches, 7-segment displays, LCD modules, and TFT screens.

B. Future Scope

Future research will expand this platform through two main directions:

- **Advanced Software Models:** Investigating complex architectures like PneuNet, Vision Transformers (ViT), and Pooling-based Vision Transformers (PiT) to boost diagnostic accuracy, accompanied by specialized HLS optimizations for self-attention memory management.
- **ASIC Implementation:** Transitioning the optimized RTL descriptions from the FPGA prototyping vehicle to custom silicon via a complete RTL-to-GDSII physical design flow, maximizing area efficiency and minimizing latency for commercial point-of-care medical devices.

IX. LIMITATIONS

Despite the successful hardware-software co-design and edge deployment of the ResNet-50 screening accelerator, several limitations should be noted:

- **Resource and Memory Constraints:** While operating within an < 8 W power budget, the Zynq-7000 FPGA has limited Block RAM and DSP slices compared to enterprise GPUs. This restricts on-chip feature caching and requires continuous data swapping via external memory interfaces.
- **Quantization Precision Trade-offs:** Converting 32-bit floating-point models to fixed-point via High-Level Synthesis introduces minor quantization loss. Aggressive bit-width reduction to conserve logic can occasionally degrade diagnostic accuracy on subtle pathologies.

- **Dataset Scope and Generalizability:** Training on curated repositories (e.g., Kermany et al.) can lead to domain shift when models encounter clinical scans from heterogeneous imaging equipment or alternative institutional protocols.
- **Throughput Constraints:** Although achieving low-latency point-of-care inference, the accelerator's raw throughput is lower than multi-GPU setups (~ 70 W+), making it optimal for real-time single-image triage rather than high-concurrency hospital-wide workflows.

DATA AND CODE AVAILABILITY

The Kaggle chest X-ray dataset is openly accessible [5]. Custom HLS and Vivado scripts are available from the corresponding author upon request.

ETHICS STATEMENT

This research used publicly available, de-identified chest X-ray datasets, exempting ethical approval since no identifiable human subjects were involved.

DECLARATION OF COMPETING INTERESTS

The authors declare no competing financial interests or personal relationships that could influence this work.

FUNDING

The authors received no financial support for the research, authorship, and/or publication of this article.

AUTHOR CONTRIBUTIONS

Gayathri Sowmya Sri Ravipati: Conceptualization, Methodology, Software (HLS implementation and Vivado hardware design), Validation, Formal analysis, Writing – Original Draft, Visualization.

Jhansi Rani Kaka: Supervision, Resources, Writing – Review & Editing, Project administration.

REFERENCES

- [1] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, K. Abulhassan, N. Al-Emadi, M. A.-A. Mohammed, M. U. Khan, "Reliable tuberculosis detection using chest x-ray with deep learning, segmentation and visualization," IEEE Access, vol. 8, pp. 191586–191601, 2020, doi:10.1109/ACCESS.2020.3031384.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc.

- IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778, doi:10.1109/CVPR.2016.90.
- [3] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, S. Xie, “ConvNeXt V2: Co-designing and scaling ConvNets with masked autoencoders,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2023, pp. 16133–16142, doi:10.1109/CVPR52729.2023.01548.
- [4] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, “Swin transformer: Hierarchical vision transformer using shifted windows,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2021, pp. 10012–10022, doi:10.1109/ICCV48922.2021.00986.
- [5] D. S. Kermany, M. Goldbaum, K. Zhang, J. Cai, F. Wang, H. Dong, X. Yuan, M. Cao, C. Ji, X. Zhang, “Identifying medical diagnoses and treatable diseases by image-based deep learning,” *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018, doi:10.1016/j.cell.2018.02.010.
- [6] P. Guddati, S. Dash, R. K. Tripathy, “FPGA implementation of the proposed DCNN model for detection of tuberculosis and pneumonia using CXR images,” *IEEE Embedded Syst. Lett.*, vol. 16, no. 4, pp. 445–448, 2024, doi:10.1109/LES.2024.3370833.
- [7] M. Vardhana, R. Pinto, J. Shen, “FPGA-based pulmonary disease detection using lightweight CNN,” *IEEE Sensors Lett.*, vol. 9, no. 10, pp. 1–4, 2025, doi:10.1109/LSSENS.2025.3604581.
- [8] F. G. Zacchigna, “Methodology for CNN implementation in FPGA-based embedded systems,” *IEEE Embedded Syst. Lett.*, vol. 15, no. 2, pp. 85–88, 2023, doi:10.1109/LES.2023.3243121.
- [9] F. Fahim, B. Hawks, C. Herwig, V. Loncar, J. Ngadiuba, N. Tran, T. Aarrestad, J. Duarte, P. Harris, S. Hauck, et al., “hls4ml: An open-source codesign workflow to empower scientific low-power machine learning devices,” arXiv preprint arXiv:2103.05579, 2021.
- [10] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, “CBAM: Convolutional block attention module,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 3–19.
- [11] S. Roy, A. Suresh, P. Sahu, A. Roy, T. R. Gupta, “Novel pooling-based VGG-Lite for pneumonia and COVID-19 detection from imbalanced chest x-ray datasets,” *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 9, no. 6, pp. 4231–4245, 2025, doi:10.1109/TETCI.2025.3577509.
- [12] K. Simonyan, A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.
- [13] M. Tan, Q. V. Le, “EfficientNetV2: Smaller models and faster training,” in Proc. Int. Conf. Mach. Learn. (ICML), 2021, pp. 10096–10106.
- [14] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” arXiv preprint arXiv:1704.04861, 2017.
- [15] A. Naveen and K. J. Rani, “Multi Class Classification of Respiratory Diseases Based on a Stacked Ensemble of AlexNet and InceptionV3 Using Chest X-Ray Images,” *International Journal of Creative Research Thoughts*, Aug. 2025.