Performance Evaluation of a Heterogeneous CPU-GPU Framework for Real-Time ECG Analytics

Leveraging OpenMP on CPUs and CUDA CNNs on NVIDIA A30 for Cardiac Signal Classification

Ariana Howell, La Chiara Landrum, Naw Safrin Sattar
Computer Science and Data Science Department, School of Applied Computational Sciences, Meharry Medical College
Nashville, Tennessee, USA.
Email: (ariana.howell, llandrum24, nawsafrin.sattar)@mmc.edu

***Abstract*—Real-time cardiac digital twins require analytics pipelines that are both computationally efficient and diagnostically accurate. This paper presents a heterogeneous high-performance computing framework integrating OpenMP-parallelized CPU preprocessing with a CUDA-accelerated convolutional neural network for ECG beat classification using the MIT-BIH Arrhythmia Database. The preprocessing pipeline achieved a 4.85× speedup and 79% runtime reduction using eight CPU threads. The GPU classifier achieved 78.17% accuracy, a macro ROC–AUC of 0.773, and throughput of 175,712 beats/s, representing an 11.2× improvement over CPU batch inference. A Digital Twin Readiness Assessment Framework evaluated predictive performance, computational efficiency, scalability, and deployment feasibility. Results indicate that preprocessing and inference satisfy real-time requirements, while classification of minority arrhythmia classes remains the primary limitation. Future work will focus on optimizing the classification model to improve minority-class performance and overall digital twin readiness.**

***Index Terms*—digital twin, ECG classification, heterogeneous computing, OpenMP, CUDA, strong scaling, weak scaling, convolutional neural network**

I. Introduction and Motivation

Digital twins for cardiac monitoring must ingest, clean, and classify ECG beats fast enough to stay synchronized with a patient while meeting the accuracy bar required for clinical decision support [1]. Signal preprocessing over multi-hour recordings is computationally heavy [2], and deep learning classifiers capable of resolving subtle arrhythmia morphologies require dedicated accelerator hardware, making a heterogeneous compute strategy a natural fit for meeting both demands at once [3].

II. Problem Statement

Existing ECG analytics research rarely integrates model optimization, heterogeneous HPC benchmarking, and scalability analysis into one workflow, making it hard to judge whether a pipeline is deployable in a real-time digital twin setting [4]. This project investigates whether OpenMP-parallelized CPU preprocessing paired with GPU-accelerated deep learning classification can deliver the throughput, latency, and accuracy profile a cardiac digital twin would require, using the MIT-BIH Arrhythmia Database as the benchmark corpus [5], [6].

III. Methodology

The CPU stage is a parallel workload: each of 48 MIT-BIH records is decoded and windowed independently, and once R-peak locations are known, the 125,556 beat windows share no state across threads. A single parallel-for directive statically distributes beat-window extraction and normalization across the thread team, while record I/O remains sequential. This logic ran once single-threaded as a baseline and again with 1, 2, 4, and 8 OpenMP threads, each averaged over 10 trials, to characterize strong scaling (fixed workload) and weak scaling (6 records per thread).

The GPU stage trains a compact one-dimensional CNN in PyTorch (44,293 parameters) on 187-sample normalized beat windows, mapping each to one of five AAMI EC57 superclasses (N, S, V, F, Q) [7]. Training ran for 8 epochs on an NVIDIA A30 via CUDA with checkpointing on validation loss. To test whether GPU acceleration is worth its cost at every batch size, the identical trained weights were also benchmarked on CPU-only inference, comparing batched (256 beats/batch) and single-beat regimes.

Table I summarizes the experimental configuration.

TABLE I
Experimental Setup

Component	Configuration
Dataset	MIT-BIH Arrhythmia Database
Records / beats	48 records / 125,556 beats
Sampling frequency	360 Hz
Classes	N, S, V, F, Q (AAMI EC57)
CPU	Intel Xeon Gold 6248R (24 cores)
CPU threads tested	1, 2, 4, 8
CPU framework	OpenMP (C++)
GPU	NVIDIA A30
GPU framework	PyTorch 2.1.3 / CUDA 13.0
CNN parameters	44,293

IV. Results

Strong-scaling preprocessing time fell from 0.456 s at one thread to 0.094 s at eight threads, a 4.85× speedup and 79% runtime reduction, but parallel efficiency declined from 1.00

to 0.61 as synchronization and bandwidth contention grew with thread count. Weak scaling showed the same signature: efficiency held near 1.00 to 2 threads before falling to 0.64 at 8 threads, confirming the overhead is bandwidth-bound rather than a fixed-workload artifact. The CNN reached 78.17% accuracy and macro ROC-AUC of 0.773, with strong discrimination for ventricular (0.932) and Q-class (0.975) beats but weaker separation for normal (0.477) and fusion (0.587) beats. The GPU was 11.2× faster than CPU in batched mode (175,712 vs. 15,619 beats/second) but the CPU was 1.31× faster in single-beat mode (4,062 vs. 3,111 beats/second), a reversal driven by per-call kernel-launch overhead that batching amortizes away.

V. Discussion

Throughput is not the limiting factor for real-time deployment: even single-beat GPU inference and single-threaded CPU preprocessing comfortably outpace real ECG data arrival rates. The binding constraint is classification reliability on rare but clinically significant classes, particularly N and F beats, pointing future effort toward model quality over further systems optimization. The single-beat reversal also implies GPU acceleration should be reserved for buffered inference, while a purely event-driven streaming path may not need a GPU at all [4], [3]. Against the seven readiness criteria, five are met outright and two remain the primary levers for improvement.

VI. Conclusion and Future Work

This work presented a heterogeneous CPU-GPU framework combining OpenMP-parallelized preprocessing with GPU-accelerated deep learning for ECG analytics, evaluated across strong scaling, weak scaling, and CPU-versus-GPU inference. Near-term work includes re-running a live weak-scaling sweep and CPU inference benchmark on the full 24-core node. Longer-term work includes streaming real-time acquisition, CNN quantization and pruning, multi-GPU scaling, multimodal digital twin integration, and clinical validation toward deployable cardiac digital twins.

References

[1] Y. Wu and B. Ghoraani, "Biological signal processing and analysis for healthcare monitoring," *Sensors*, vol. 22, no. 14, p. 5341, 2022. [Online]. Available: https://doi.org/10.3390/s22145341

[2] X. Dong, C. Chen, Q. Geng, W. Zhang, and X. D. Zhang, "Fast algorithm based on parallel computing for sample entropy calculation," *IEEE Access*, vol. 9, pp. 20 223–20 234, 2021. [Online]. Available: https://doi.org/10.1109/ACCESS.2021.3054750

[3] A. Kanduri, S. Shahhosseini, E. K. Naeini, H. Alikhani, P. Liljeberg, N. Dutt, and A. M. Rahmani, "Edge-centric optimization of multi-modal ML-driven eHealth applications," arXiv:2208.02597, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2208.02597

[4] L. M. Pinho, "Real-time parallel programming: State of play and open issues," arXiv:2303.11018, Mar. 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.11018

[5] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001. [Online]. Available: https://physionet.org/content/mitdb/

[6] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000. [Online]. Available: https://physionet.org/about/content/

[7] Association for the Advancement of Medical Instrumentation, "ANSI/AAMI EC57:2012/(R)2020: Testing and reporting performance results of cardiac rhythm and ST segment measurement algorithms," 2020. [Online]. Available: https://webstore.ansi.org/preview-pages/AAMI/preview$_{A}NSI+AAMI+EC57-2012+(R2020).pdf$