Cybersecure End-to-end FPGA-accelerated ECG Monitoring for Precision Diagnosis with Personalized CWT and Adversarial Defense

Tiancheng Cao, *Member, IEEE*, Wei Soon Ng, *Student Member, IEEE*, Chen Shen, *Student Member, IEEE*, Hongtao Li, Rong Tan, Dawei Wang, Hen-Wei Huang, *Member, IEEE*

***Abstract*— Advancements in wearable technology and edge computing have transformed cardiovascular monitoring, driving the demand for private, secure, and real-time diagnostic solutions. This paper presents an edge-wearable ECG monitoring system that integrates personalized continuous wavelet transform (CWT) preprocessing, a DeepFool–FGSM adversarial defense, and an optimized parallel PoolFormer architecture for resource-constrained FPGA deployment. The personalized CWT captures individual-specific ECG features and mitigates model-inversion privacy risks. The defense approach balances robustness and computational efficiency and reduces hardware complexity and energy via quantization-aware training (QAT). Evaluations on field programmable gate array (FPGA) confirm high diagnostic accuracy (98.93%), real-time inference (latency <1.7 ms), and improved robustness against adversarial perturbations, with 0.055 W FPGA-core power. Together, the system delivers confidentiality, integrity, and availability for cybersecure, personalized ECG monitoring at the edge.**

***Index Terms*—Edge Computing, Precision Diagnosis, Secure AI, Personalized Electrocardiography, FPGA**

I. Introduction

Recent progress in wearables and edge computing is transforming how we monitor cardiovascular health [1,2]. Electrocardiography (ECG) remains the gold standard for non-invasive, early detection of cardiac issues [3], but relying on remote servers for data processing raises privacy and latency concerns, especially in less connected or resource-limited settings [4].

Thanks to advances in sensor miniaturization and real-time on-device analytics, today's wearables can continuously track heart signals and support more personalized care [5]. By keeping computation at the edge, these systems reduce risks of data exposure and cyber-attacks while easing the burden on wireless networks [6]. As wearable devices become ubiquitous, there is a pressing need for efficient and secure processing solutions that can operate under stringent power and computational constraints [7,8]. Integrating advanced signal processing and lightweight machine learning models on edge platforms is key to enabling personalized, real-time diagnostics [9]. The convergence of these technologies will not only improve the accuracy of cardiovascular monitoring but also support scalable, decentralized healthcare even in resource-limited settings.

In this study, we present an edge-wearable ECG monitoring system that integrates a personalized CWT preprocessing pipeline for individual-specific spectral feature extraction, an adversarially robust DeepFool–FGSM training framework [10], and a parallel PoolFormer architecture optimized for efficient real-time inference on FPGAs. Our main contributions are:

1. A modified PoolFormer architecture, parallel PoolFormer, tailored for resource-constrained FPGA deployment. By significantly reducing the model parameter count and computational complexity, our refined architecture enables efficient real-time inference without sacrificing accuracy.
2. An adversarial training strategy using a DeepFool–FGSM defense framework to enhance model robustness. This method explicitly mitigates the risks posed by subtle input modifications, thereby reinforcing the robustness and reliability of the classification model in real-world scenarios.
3. A personalized CWT preprocessing framework that captures individual-specific ECG features while ensuring data privacy. This tailored approach not only enhances diagnostic precision but also ensures the privacy of sensitive health data.

The remainder of this paper is organized as follows: Section II reviews related work; Section III details the system architecture and parallel PoolFormer; Section IV describes quantization-aware adversarial training with the DeepFool–FGSM defense framework; Section V covers personalized CWT signal processing; Section VI presents FPGA implementation results; and Section VII concludes the paper.

II. Related Works

Over the past decade, machine learning has enabled significant advances in wearable and edge ECG monitoring for

Manuscript received MMM DD, 2025; This research is supported by Nanyang Assistant Professorship Start-up Grant and the Eric and Wendy Schmidt AI in Science Postdoctoral Fellowship, a Schmidt Futures program. *(Corresponding author: Tiancheng Cao).*

Tiancheng Cao, Wei Soon Ng, Hongtao Li, Chen Shen, Rong Tan and Dawei Wang are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798 (e-mail: tiancheng.cao@ntu.edu.sg; weisoon001@e.ntu.edu.sg; hongtao001@e.ntu.edu.sg; shenchen@ntu.edu.sg; rong.tan@ntu.edu.sg; dawei.wang@ntu.edu.sg).

Hen-Wei Huang is with the School of Electrical and Electronic Engineering and Lee Kong Chian School of Medicine, Nanyang Technological University, Singapore, 639798 (e-mail: henwei.huang@ntu.edu.sg).

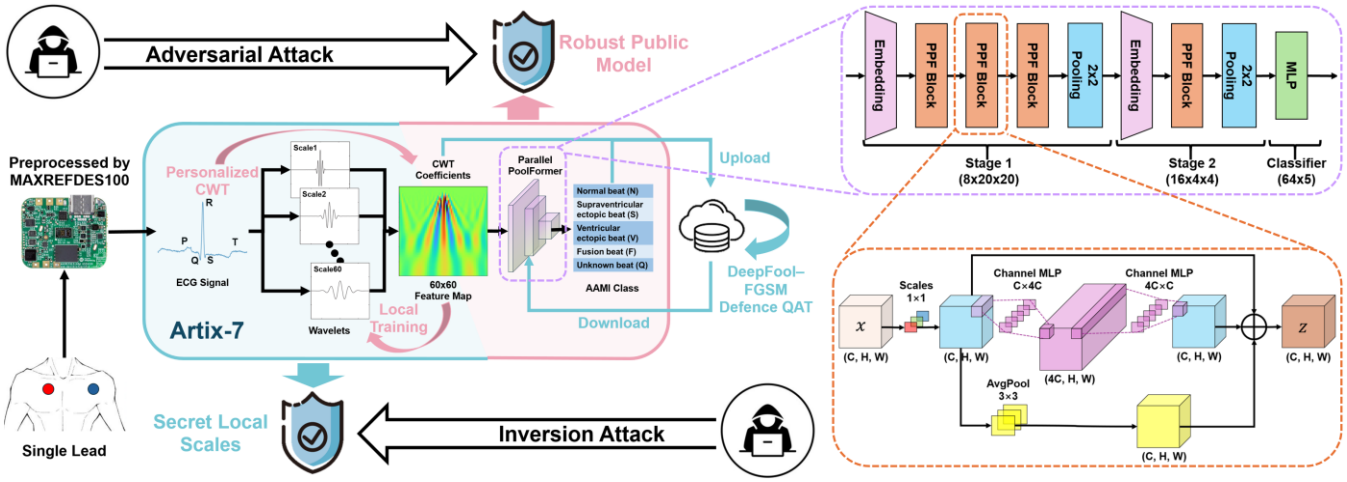


Fig. 1. Cybersecure edge-wearable ECG monitoring framework with personalized CWT preprocessing, adversarial defense, and FPGA-accelerated inference. A single-lead ECG (360 Hz) is acquired using the MAXREFDES100 board. The FPGA implements a locally personalized CWT to extract 60×60 time–frequency spectrograms, which serve as input to the parallel PoolFormer (PPF) model for AAMI-standard arrhythmia classification. The model is trained with DeepFool–FGSM adversarial defense and 8-bit quantization-aware training (QAT) for robust, efficient deployment. Privacy is further protected by secret local CWT scales, preventing inversion attacks from spectrograms.

arrhythmia detection and real-time screening [11]. Despite the shift from cloud to edge to address latency and data locality, most pipelines still use fixed preprocessing and generic models, overlooking substantial inter-individual ECG variability and thus limiting generalization [12].

Deterministic spectro-temporal transforms can retain biometric signatures, raising privacy concerns such as re-identification and linkage attacks if features leave the device boundary [13]. Federated and on-device learning reduce raw data exposure, but do not obfuscate invertible features, highlighting the need for personalized, privacy-preserving edge preprocessing [12,13].

Robustness and deployment challenges further complicate on-device ECG analysis: small perturbations can induce clinically significant misclassifications [14], and most defense strategies are developed for compute-rich settings, limiting practical deployment on resource-constrained wearables [15]. Additionally, existing systems rarely unify personalized feature extraction, adversarial robustness, and hardware efficiency [16,17]. These gaps motivate an integrated on-device framework addressing personalization, privacy, robustness, and efficiency.

III. Edge ECG Monitoring System

A. Overall Architecture

The proposed edge-wearable ECG monitoring framework (Fig. 1) integrates three key components: personalized continuous wavelet transform (CWT) preprocessing, DeepFool–FGSM defense, and FPGA-accelerated deployment. Raw ECG signals are acquired from wearable sensors and locally transformed by a personalized CWT module, where user-specific scale sets are stored on-device. This design adapts to individual ECG morphology, extracts discriminative time–frequency features, and simultaneously preserves privacy by reducing the risk of reconstructing raw signals from released features.

At the system core, a quantized parallel PoolFormer model pre-trained on the MIT-BIH database (48 records from 47 subjects at 360 Hz, MLII) [18] performs arrhythmia classification. To enhance robustness under hardware constraints, the model is trained with a proposed hybrid DeepFool–FGSM defense and quantization-aware training (QAT). Finally, the entire framework is deployed on an Artix-7 100T FPGA through hardware–software co-design, jointly optimizing model architecture and implementation for lightweight, energy-efficient, and secure real-time ECG monitoring.

B. Beat Segmentation and Data Augmentation

During the pre-training, each ECG beat is segmented around the R-wave peak to ensure precise isolation of individual heartbeats. The window of a single beat is defined by:

$$T(Rpeak(k-1)+b) \le T\big(Rpeak(k)\big) \le T(Rpeak(k+1)-(120-b)) \quad (1)$$

where $T(Rpeak(k))$ is the R-wave peak time of annotation k and b is the beat range bias. The constant 120 represents the excluded margin before and after the target beat within sample window, preventing overlap from adjacent R-peaks and ensuring consistent segmentation. To mitigate class imbalance and improve generalization, we adopt biased random sampling as shown in Fig. 2, epoch-wise subsampling of majority classes to 10k samples with random-biased sample shifting augmentation for minority classes to 4k samples, followed by a small zero-mean noise injection as a label-preserving perturbation to training beats.

C. Parallel PoolFormer Architecture

Building on previous modifications [19], the PoolFormer architecture is further streamlined by reducing the number of stages and replacing normalization with a simplified scaling

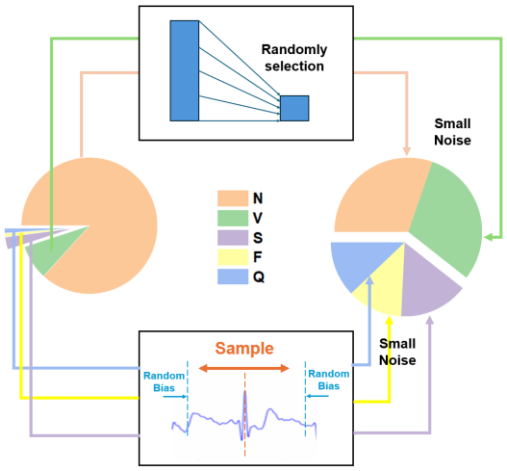


Fig. 2. Biased random sampling with epoch-wise majority subsampling, minority random-biased shifting, and small label-preserving zero-mean noise. (Patient-wise 80/10/10 for training/validating/testing)

layer, thereby enhancing inference efficiency without compromising performance. Inspired by recent advances in large language model architectures [20], we propose a parallel PoolFormer block to improve computational parallelism and reduce resource overhead. The conventional two-stage PoolFormer block [17] can be formulated as

$$y = x + pool(scale_1(x)) \tag{2}$$

$$z = y + MLP(scale_2(y)) \tag{3}$$

where $scale(\cdot)$ is the scaling layer, $pool(\cdot)$ is the average pooling and $MLP(\cdot)$ is the channel multi-layer perceptron which has a multiple ratio of 4. To minimize the sequential computation and intermediate variables, we leverage the linearity and distributive properties of these operations to merge consecutive linear transformations. This allows the second stage to be unfolded and simplified as a single trainable linear transformation:

$$MLP\big(scale_2(y)\big)$$

$$= MLP\big(scale_2(x) + scale_2(pool(scale_1(x)))\big) \tag{4}$$

$$= MLP(linear'(x)) \tag{5}$$

Because both the identity mapping and learnable scaling are channel-wise linear transforms, enabling their fusion into a single learnable channel scaling $scale'(\cdot)$. The final parallel PoolFormer block thus computes:

$$z = x + pool\big(scale_1(x)\big) + MLP(linear'(x))$$

$$= scale'(x) + pool\big(scale'(x)\big) + MLP'(scale'(x)) \tag{6}$$

By exploiting the linear characteristics of its operations and learnable parameters, the block can be rearranged into a parallel format as shown in Fig. 1. This reorganization simplifies the structure, enhances parallelizability, and preserves skip/merge functionalities. Experimentally, the parallel block reduces memory usage by over 25% and significantly improves inference speed and energy efficiency, demonstrating its effectiveness for edge deployment. This streamlined memory usage reduces both data transfer and computation, resulting in substantial improvements in inference speed and energy efficiency.

IV. DeepFool–FGSM Defense Framework

To strengthen system robustness under adversarial and low-precision constraints, we propose a hybrid defense that combines DeepFool and FGSM adversarial training with Quantization-Aware Training (QAT). This framework defends against both gradient-based (PGD [21], BIM [22]) and decision-based (HSJ [23]) attacks, while ensuring robustness at deployment precision on resource-constrained edge devices.

A. DeepFool Sample Generation with FGSM Training

DeepFool generates adversarial examples by iteratively linearizing decision boundaries to estimate the minimal perturbation required to alter a prediction. This iterative process produces boundary-sensitive perturbations that remain close to the original data manifold, thereby revealing vulnerabilities in regions critical to classification. Training with such finely tuned samples strengthens the model against subtle, hard-to-detect adversarial attacks. In contrast, FGSM applies a one-step gradient update to perturb inputs along the most vulnerable direction. Although less precise than DeepFool, FGSM is highly efficient and can generate a large number of adversarial samples with minimal overhead. Its broad but coarser perturbations complement the fine-grained adversarial signals of DeepFool.

By combining the two, we establish the proposed DeepFool–FGSM defense framework, where DeepFool provides finely tuned adversarial sample generation that probes decision boundaries, and FGSM injects efficient one-step perturbations directly into the training loop. This division of roles allows the framework to balance robustness and efficiency, ensuring the model is simultaneously exposed to boundary-level perturbations and fast, computationally lightweight adversarial examples, thereby improving both security and deployability on edge devices.

B. Quantization-Aware Training

To accommodate the constraints of edge wearable devices, Quantization-Aware Training (QAT) is integrated into the adversarial training pipeline. QAT simulates low-precision arithmetic during training, ensuring the model maintains robustness when deployed on hardware with limited numerical precision. We employ dynamic weight quantization, truncating the extreme 2% of the weight distribution at both ends [7], which normalizes weights, reduces quantization noise, and preserves accuracy and adversarial robustness even at W8/A16 precision in pre-FPGA tests.

During training, as illustrated in Fig. 3, the model receives both original and adversarial samples. In the forward pass, INT8 quantization is applied to the weights in the parallel PoolFormer, closely mimicking deployment conditions. Gradients are computed with respect to the original floating-point weights in the backward pass, preserving optimization fidelity. This software-hardware co-design harmonizes

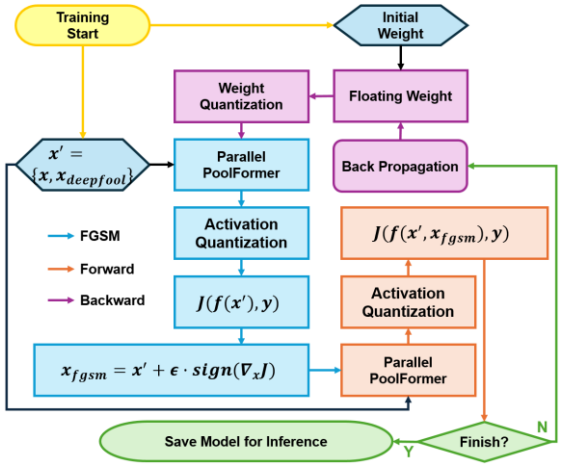


Fig. 3. Training flowchart of the DeepFool–FGSM adversarial defense with quantization-aware training. This framework enhances model robustness and hardware deployment efficiency.

adversarial robustness with hardware-aware optimization, enabling secure, efficient, and real-time inference on resource-constrained edge devices.

C. Framework Performance Analysis

We evaluated the DeepFool–FGSM defense framework on a parallel PoolFormer network with parameters in TABLE I, subjected to three attack types: PGD, BIM, and HSJ. Model performance was assessed at three stages: (i) baseline without defense, (ii) DeepFool–FGSM defense in full precision, and (iii) DeepFool–FGSM defense with INT8 QAT.

As shown in Fig. 4(a), the baseline model's accuracy dropped sharply in the presence of adversarial attacks, clearly revealing its vulnerability. However, after integrating our DeepFool–FGSM defense framework, the model's robustness improved markedly across all tested attacks shown in Fig. 4(b). This gain reflects the complementary strengths of DeepFool's precise, boundary-focused perturbations and FGSM's efficient, broad coverage during adversarial training. The addition of QAT further maintained this robustness under low-precision (INT8) deployment, with the co-design approach reducing computational complexity and energy consumption while preserving defense integrity.

These results demonstrate that combining adversarial training with quantization-aware optimization not only strengthens model security against attacks but also ensures efficient and reliable deployment on edge wearable devices, bridging the gap between high-performance AI and practical hardware constraints.

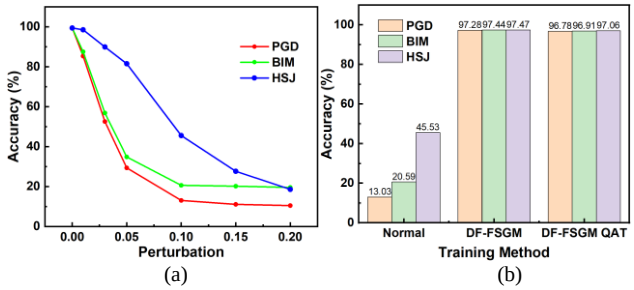
V. Personalized CWT Preserving Data Privacy

A. Motivation for Localized Preprocessing

While adversarial training can enhance neural network robustness against input perturbations, deep learning models in healthcare remain vulnerable to model inversion attacks, which can reconstruct original biometric signals by exploiting model parameters and outputs, thus threatening privacy [24]. Conventional preprocessing pipelines, which use fixed, global transformations, further exacerbate this risk by generating

TABLE I.
PARALLEL-POOLFORMER HYPER-PARAMETERS

Stage	# Token	Layer Specification		Parallel PoolFormer
1	$\frac{H}{3} \times \frac{W}{3}$	Patch Embed.	Patch Size	3×3, stride 3
			Embed. Dim.	8
		PPF Block	Pooling Size	3×3, stride 1
			MLP Ratio	4
			# Block	3
2	$\frac{H}{15} \times \frac{W}{15}$	Patch Embed.	Patch Size	3×3, stride 3
			Embed. Dim.	16
		PPF Block	Pooling Size	3×3, stride 1
			MLP Ratio	4
			# Block	1
Parameters (k)				5.4
MACs (M)				0.71



(a) (b)

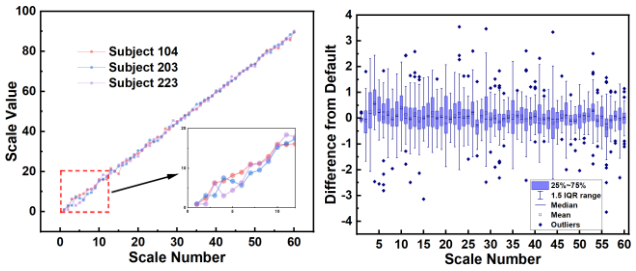
Fig. 4. (a) Accuracy of the baseline model under different perturbation levels when subjected to adversarial attacks (PGD, BIM, and HSJ). (b) Classification accuracy under attacks across different training methods.

deterministic intermediate features that retain individual identity information. As a result, static transforms can inadvertently enable user re-identification or spectrogram inversion attacks.

To address this, we propose an on-device, personalized CWT preprocessing framework that stores both raw ECG data and subject-specific scales locally. By adapting scale parameters through local training to each individual's ECG characteristics, our approach not only obfuscates biometric identity and reduces wireless payload, but also optimizes diagnostic performance. This decentralized design both mitigates feature invertibility and ensures spectral representations align with subject-specific patterns, thereby enhancing privacy and signal fidelity.

B. Personalized Continuous Wavelet Transform

The continuous wavelet transform (CWT) decomposes a signal into time–frequency components at multiple scales, providing a multi-resolution analysis well-suited for non-stationary signals such as ECG, where features like QRS complexes and P/T waves vary over time. Unlike the Short-Time Fourier Transform (STFT), which uses fixed window sizes and faces a trade-off between time and frequency resolution, CWT adapts to both fast and slow signal dynamics. Our hardware-optimized edge implementation employs a discrete, fixed-point streaming version of CWT, supporting real-time processing and low power consumption while keeping raw ECG data local and reducing wireless bandwidth by transmitting only compact, task-specific features [25]. Based on this, the default-mode representation for our personalized implementation can be calculated as:



(a) (b)

Fig. 5. (a) Comparison of personalized CWT scale profiles from three representative subjects (104, 203, 223) against the default linear scale distribution, with zoom-in highlighting individual variability at low scales. (b) Box plot showing the distribution of scale deviations across all subjects.

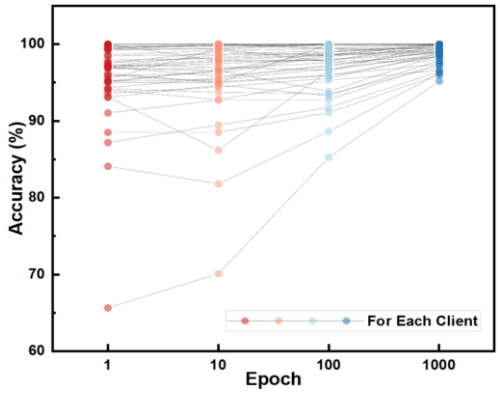


Fig. 6. Classification accuracy across subjects over local training epochs for personalized CWT.

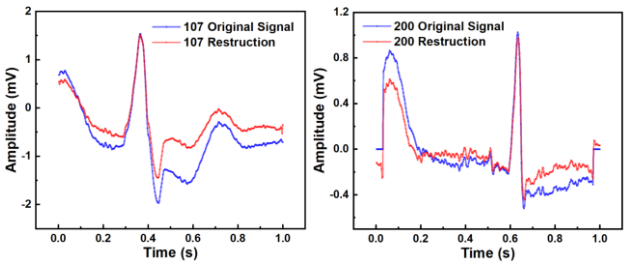
$$CWT\ (j,k) = \sum_n f[n] \cdot \varphi[n-2k]$$

$$= \frac{1}{\sqrt{j}} \sum_n f[n] \cdot \cos\left(\omega \cdot \frac{n-2k}{j}\right) \cdot e^{-\frac{\left(\frac{n-2k}{j}\right)^2}{2}} \qquad (7)$$

where, $f[n]$ represents the discrete signal at sample n, $\varphi[n]$ is the wavelet function and j and k are the scale and translation of the transform, respectively.

In this study, we set the default 60 CWT scales linearly from 1 to 90 and use the Morlet wavelet as the basis function. On the edge device, local training with each individual's ECG data is used to adapt these scales, ensuring all preprocessing remains subject-side. Analysis of personalized CWT scale profiles across multiple subjects shown in Fig. 5 reveals substantial inter-subject variability, demonstrating that a default approach is insufficient and highlighting the need for personalized preprocessing to achieve accurate ECG analysis. Our experiments on the MIT-BIH dataset, covering 47 subjects with diverse arrhythmia types, confirm this variability and support the broad applicability of our approach.

By tailoring the CWT scales to each individual's ECG through local training, the transform captures clinically important features, such as the specific morphologies of QRS complexes, P waves, and T waves, while suppressing noise and irrelevant spectral components. This personalized adaptation improves feature separability, enabling more effective classification by deep learning models. As shown in Fig. 6, classification accuracy increased progressively as CWT scales



(a) (b)

Fig. 7. Reconstructed ECG signal sample of (a) 107 and (b) 200 subject from spectrograms using default CWT scales show significant deviation from the original signals

TABLE II.
COMPARISON OF SIGNAL RECOVERABILITY

	RMSE	SNR(dB)	ID Classification
w/ Personalized CWT	0.45 ± 0.26	1.88 ± 1.11	64.80 %
w/o Personalized CWT	0.072 ± 0.043	17.87 ± 1.65	95.72 %

were locally optimized for each subject, leading to more robust and precise diagnostic results.

C. Identification protection

Multiple studies have shown that raw ECG waveforms can uniquely identify individuals, posing significant privacy risks if signals are reconstructed from released features [26]. Our personalized CWT preprocessing inherently mitigates this threat: because the individualized scale parameters remain undisclosed, adversaries cannot accurately invert the spectrogram to recover the original ECG, even if the features are intercepted. As Fig. 7 demonstrates, attempts to invert spectrograms using default scales yield reconstructions that differ markedly from true waveforms.

Table II compares reconstruction metrics (RMSE, SNR) and re-identification results obtained using a 2D CNN classifier, with and without personalized CWT scales. When personalized, the quality of signal inversion and identity classification drops substantially, indicating reduced risk of model inversion attacks. In contrast, using default linear scales enables much more accurate reconstruction and identification, highlighting a significant privacy vulnerability. While this study focuses on feature-level protection, conventional encryption can be used in parallel to secure data in transit; our personalized preprocessing further safeguards privacy even if communication channels are compromised.

VI. FPGA IMPLEMENTATION RESULTS AND DISCUSSION

To ensure availability, the FPGA pipeline sustains real-time latency and low power. The framework was first trained offline on an i5-13600KF CPU and RTX3060 GPU, achieving 98.93% diagnostic accuracy across all subjects. Benchmark comparisons in Table III show that our method delivers competitive accuracy with the smallest model size, full hardware support, personalized preprocessing, and cybersecurity. In contrast, traditional non–deep-learning approaches either yield lower accuracy or rely on complex

TABLE III.
COMPARISON BENCHMARK ACCURACY

Work	JBHI 2022 [27]	IEEE Access 2021 [28]	JBHI 2024 [29]	TBio-CAS 2022 [30]	This Work
Feature Extrac.	N.R.	Template Match	N.R.	LC Sampling	CWT
Model Struct.	CNN	Template +FSM	ResNet	Spiking rMLP	Parallel Pool-Former
Model Size (k)	8.2	**3.2**	90	14.3	5.4
# Classes	5	2	**6**	5	5
Acc.	**0.991**	0.981	N.R.	0.982	0.989
Recall	N.R.	N.R.	N.R.	0.98	**0.986**
Precision	N.R.	N.R.	N.R.	0.983	**0.984**
F1 Score	N.R.	N.R.	0.982	0.982	**0.985**
Hardware	MCU	FPGA	No	**FPGA+ ASIC**	FPGA
Power (W)	0.026*	0.081	N.R	**9.3e-7**	0.055
Latency (ms)	27	910	N.R	**0.50**	1.66
Energy (mJ)	0.71	73.7*	N.R	**7.5e-4**	0.091
Person-alized Diagnosis	No	Yes	Yes	No	**Yes**
Cyber-security	No	No	No	No	**Yes**

N.R.: Not Reported
**Calculated from values reported in the original paper.*

ensembles impractical for edge deployment [31]. The optimized model was then deployed on an Artix-7 FPGA, where hardware acceleration enabled real-time ECG processing and diagnosis. The deployment integrates three co-designed modules: (i) a personalized CWT preprocessor, (ii) a quantized parallel PoolFormer inference engine, and (iii) an adversarial defense layer using DeepFool–FGSM with QAT. Unified hardware–software co-optimization ensures real-time, energy-efficient, and privacy-preserving inference suitable for edge wearables.

A. Hardware Architecture

The CWT and Parallel PoolFormer implemented on the FPGA using a streaming architecture [32], where each layer is mapped to a dedicated hardware block. These blocks operate independently and communicate via a custom ready/valid handshake protocol, forming a pipelined architecture. All hardware components are designed at the RTL level using a hardware description language (HDL), as detailed in [33]. The CWT layer is implemented as a 1D convolution with 60 output channels and reuses the same hardware architecture as the standard convolutional layers. The fusion of convolution and average pooling further facilitates a software–hardware co-design. This implementation framework enables rapid integration and deployment of the neural network (NN) on the FPGA, delivering high throughput with optimal resource consumption. The high-level hardware architecture implemented mirrors the Parallel PoolFormer module

TABLE IV.
FPGA IMPLEMENTATION DETAILS

Development Board		Arty A7 100T
FPGA		XC7A100T
Feature Extraction		Personalized CWT
NN Implemented		Parallel PoolFormer
Model Size (Kbit)		43.2
CWT	**Precision**	W16/A16
	LUTs	11413
	FFs	15496
	BRAMs	17.5
Parallel PoolFormer	**Precision**	W8/A16
	LUTs	20794
	FFs	31498
	BRAMs	72
Inference Latency (ms)		1.66
Inference Power (W)	**Dev. Board**	1.50
	FPGA core	0.055
Energy/Inference (mJ)	**Dev. Board**	2.49
	FPGA core	0.091

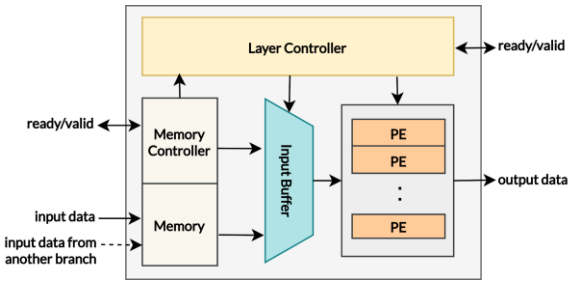


Fig. 8. Block-level hardware architecture of a hardware layer. Each layer operates independently and communicates with adjacent layers using a custom ready/valid handshake protocol.

illustrated in Fig. 1, with each convolutional/pooling layer instantiated as a distinct hardware block, following the block-level architecture shown in Fig. 8.

B. Measurement Results

The hardware design is synthesized and implemented in default mode using AMD Xilinx Vivado 2022.1.2, and the generated bitstream is used to program the Artix-7 FPGA (XC7A100T). Implementation details are summarized in Table IV. The personalized CWT and Parallel PoolFormer modules are integrated into a unified system, with the CWT block allocated higher weight precision to preserve inference accuracy. The final hardware implementation maintains identical accuracy to the software model, as all quantization parameters are pre-validated through simulation prior to deployment. Resource utilization is reported from the post-implementation results generated by Vivado. The measurement set is shown in Fig. 9 and methodologies for the remaining evaluation metrics are described below:

- *Inference Latency*: Input data batches are transmitted from a host computer to the FPGA via UART. The average inference latency is computed based on the number of inference results returned within a fixed time window. This approach minimizes the impact of UART-induced latency. Given that the total number of computation cycles for the CWT and Parallel

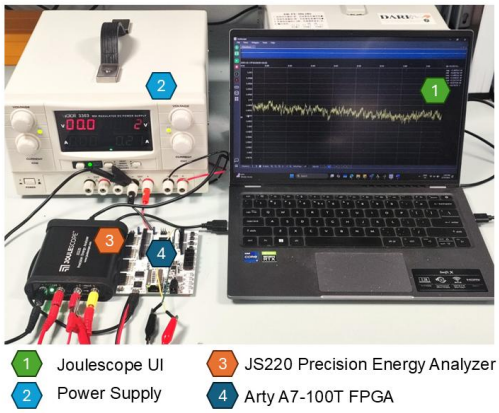


Fig. 9. Power consumption measurement setup with Power JS220 Precision Energy Analyzer.

PoolFormer is deterministic, the average inference latency accurately represents the system's inference latency.

- *Average Inference Power*: The Arty7 100T development board is powered by a power supply through a JS220 precision power analyzer. Average power consumed by the development board is measured while the development board executes inference workloads. The FPGA core power is obtained using the built-in power monitoring rail, as described in the Arty7 manual.
- *Energy per Inference*: This metric is computed as the product of average power and inference latency. It serves as a key indicator of the energy efficiency of the implemented system.

With an inference latency well below 2*ms*, the system is suitable for real-time applications. It is important to note that the reported power reflects the total consumption of the entire development board, including unused components that may contribute to power consumption. Therefore, the actual power consumption of the FPGA fabric alone is expected to be significantly lower. Furthermore, due to the reconfigurable nature of FPGA platforms, both the CWT and Parallel PoolFormer modules can be easily fine-tuned or updated as needed. This flexibility enhances the system's adaptability and can contribute to increased robustness against adversarial attacks.

Compared with existing benchmarks for 5-class classification shown in Table III, our implementation achieves significantly lower latency and reduced energy per inference than the MCU-based implementation. Although our hardware design does not surpass the latency and energy efficiency of the ASIC-based implementation reported in [42], the FPGA platform offers greater robustness, flexibility, and practicality, which are critical for secure and personalized medicine applications.

Beyond FPGAs, the architecture is also compatible with ASICs, which can benefit from hardwired data paths and tighter memory-compute integration, leading to improved energy and area efficiency. However, this comes at the cost of reconfigurability, limiting post-deployment updates and security enhancements. The choice to adopt heterogeneous platforms or remain on standalone FPGA/ASIC implementations should depend on the specific target application requirements and deployment conditions.

C. Discussion

In this study, we developed an edge-wearable ECG monitoring system that integrates personalized CWT for individualized feature extraction, a DeepFool–FGSM adversarial defense, and a parallel PoolFormer architecture optimized for FPGA deployment. The personalized CWT not only strengthens protection of biometric data but also enhances feature separability. The adversarial training framework mitigates misclassification risks under attacks, while the streamlined PoolFormer architecture reduces model parameters and power consumption, enabling real-time inference on resource-constrained devices. Together, these findings highlight the feasibility of secure and efficient cardiovascular monitoring at the edge.

Despite these advantages, the study has limitations. Evaluation was confined to a single FPGA platform, the MIT-BIH database with limited diversity, and a restricted set of adversarial scenarios. Broader validation across heterogeneous datasets and devices will be necessary to confirm generalizability. Future work will include physiologically constrained synthetic beats to address class imbalance, transfer learning from larger ECG databases to improve out-of-distribution robustness, and wider hardware benchmarking to enhance portability.

Clinical translation is also a key next step. Collaborations with hospital partners will allow validation on more diverse patient populations and quantitative assessment of demographic variability. While this work did not focus on explicit interval extraction (e.g., RR, ST, QT), the preserved waveform information enables extension toward automated clinical indices and individualized therapy planning. Moving beyond beat-level classification, the framework can also be adapted for event-level detection of arrhythmias such as atrial fibrillation and ischemia.

Finally, the system’s low-latency and privacy-preserving features suggest applicability beyond remote monitoring, including acute hospital settings such as Emergency Department triage. Integration with compliance teams will further ensure alignment with regulatory and governance requirements.

VII. Conclusion

In conclusion, this study presents an edge-wearable ECG monitoring framework that integrates personalized CWT preprocessing, a DeepFool–FGSM adversarial defense, and a parallel PoolFormer optimized for FPGA deployment. Confidentiality, integrity, and availability are jointly achieved through local feature personalization, robustness against PGD/BIM/HSJ attacks at deployment precision, and real-time low-power inference. The framework achieved high diagnostic accuracy, effective protection against biometric re-identification, and was validated on FPGA for feasibility in resource-constrained environments. Future investigations should focus on broader hardware evaluation and clinical validation to support widespread adoption in secure and personalized cardiovascular diagnostics.

ACKNOWLEDGMENT

The authors acknowledge X. Xu for contributions to the preliminary experiments.

REFERENCES

[1] K. C. Siontis, P. A. Noseworthy, Z. I. Attia, *et al.*, "Artificial intelligence-enhanced electrocardiography in cardiovascular disease management," *Nat. Rev. Cardiol.*, vol. 18, pp. 465–478, July 2021, doi: 10.1038/s41569-020-00503-2.

[2] V. Sangha, B. J. Mortazavi, A. D. Haimovich, *et al.*, "Automated multilabel diagnosis on electrocardiographic images and signals," *Nat. Commun.*, vol. 13, Art. no. 1583, Mar. 2022, doi: 10.1038/s41467-022-29153-3.

[3] T. Cao, Z. Zhang, W. L. Goh, C. Liu, Y. Zhu, and Y. Gao, "ECG classification using binary CNN on RRAM crossbar with nonidealities-aware training, readout compensation and CWT preprocessing," *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Toronto, ON, Canada, 2023, pp. 1-5, doi: 10.1109/BioCAS58349.2023.10389002.

[4] R. Wang, J. Lai, Z. Zhang, X. Li, P. Vijayakumar, and M. Karuppiah, "Privacy-preserving federated learning for Internet of Medical Things under edge computing," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, pp. 854-865, Feb. 2023, doi: 10.1109/JBHI.2022.3157725.

[5] A. Libanori, G. Chen, X. Zhao, *et al.*, "Smart textiles for personalized healthcare," *Nat. Electron.*, vol. 5, pp. 142–156, Mar. 2022, doi: 10.1038/s41928-022-00723-z.

[6] Y. R. Siwakoti, M. Bhurtel, D. B. Rawat, A. Oest, and R. C. Johnson, "Advances in IoT security: Vulnerabilities, enabled criminal services, attacks, and countermeasures," *IEEE Internet Things J.*, vol. 10, no. 13, pp. 11224-11239, Jul. 2023, doi: 10.1109/JIOT.2023.3252594.

[7] T. Cao *et al.*, "A non-idealities aware software–hardware co-design framework for edge-AI deep neural network implemented on memristive crossbar," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 4, pp. 934-943, Dec. 2022, doi: 10.1109/JETCAS.2022.3214334.

[8] A. Lakhan *et al.*, "Restricted Boltzmann machine assisted secure serverless edge system for Internet of Medical Things," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, pp. 673-683, Feb. 2023, doi: 10.1109/JBHI.2022.3178660.

[9] T. Cao, C. Liu, Y. Gao, and W. L. Goh, "Parasitic-aware modeling and neural network training scheme for energy-efficient processing-in-memory with resistive crossbar array," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 2, pp. 436-444, June 2022, doi: 10.1109/JETCAS.2022.3172170.

[10] S. M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 2574-2582.

[11] T. Cao, H. Li, and H.-W. Huang, "LiMO: A lightweight MambaOut for end-to-end IoMT ECG diagnosis with fully configurable quantization co-design on MCU," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2025, *accepted for publication*.

[12] M. Hammad, M. ElAffendi, and A. A. Abd El-Latif, "CardioECGNet: A novel deep learning architecture for accurate and automated ECG signal classification across diverse cardiac conditions," *Biomed. Signal Process. Control*, vol. 106, Art. no. 107720, Aug. 2025, doi: 10.1016/j.bspc.2025.107720.

[13] G. Zheng *et al.*, "Finger-to-Heart (F2H): Authentication for wireless implantable medical devices," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 4, pp. 1546-1557, July 2019, doi: 10.1109/JBHI.2018.2864796.

[14] H. Chen, C. Huang, Q. Huang, Q. Zhang, and W. Wang, "ECGadv: Generating adversarial electrocardiogram to misguide arrhythmia classification system," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, pp. 3446-3453, 2020, doi: 10.1609/aaai.v34i04.5748.

[15] T. Cao, Z. Zhang, W. L. Goh, C. Liu, Y. Zhu, and Y. Gao, "A ternary weight mapping and charge-mode readout scheme for energy-efficient FeRAM crossbar compute-in-memory system," *Proc. IEEE 5th Int. Conf. Artif. Intell. Circuits Syst. (AICAS)*, Hangzhou, China, 2023, pp. 1-5, doi: 10.1109/AICAS57966.2023.10168639.

[16] G. Sivapalan, K. K. Nundy, S. Dev, B. Cardiff, and D. John, "ANNet: A lightweight neural network for ECG anomaly detection in IoT edge sensors," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 1, pp. 24-35, Feb. 2022, doi: 10.1109/TBCAS.2021.3137646.

[17] T. Cao, W. Yu, Y. Gao, C. Liu, S. Yan, and W. L. Goh, "RRAM-PoolFormer: A resistive memristor-based PoolFormer modeling and training framework for edge-AI applications," *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Monterey, CA, USA, 2023, pp. 1-5, doi: 10.1109/ISCAS46773.2023.10181612.

[18] 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.

[19] T. Cao *et al.*, "Edge PoolFormer: Modeling and training of PoolFormer network on RRAM crossbar for edge-AI applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 33, no. 2, pp. 384-394, Feb. 2025, doi: 10.1109/TVLSI.2024.3472270.

[20] B. He and T. Hofmann, "Simplifying transformer blocks", *arXiv preprint arXiv:2311.01906*, 2023.

[21] H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann, and M. Unser, "CNN-based projected gradient descent for consistent CT image reconstruction," *IEEE Trans. Med. Imaging*, vol. 37, no. 6, pp. 1440-1453, June 2018, doi: 10.1109/TMI.2018.2823325.

[22] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[23] J. Chen, M. I. Jordan, and M. J. Wainwright, "HopSkipJumpAttack: A query-efficient decision-based attack," *Proc. IEEE Symp. Security Privacy (SP)*, San Francisco, CA, USA, 2020, pp. 1277-1294, doi: 10.1109/SP40000.2020.00045.

[24] J. S. Arteaga-Falconi, H. Al Osman, and A. El Saddik, "ECG authentication for mobile devices," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 3, pp. 591-600, Mar. 2016, doi: 10.1109/TIM.2015.2499018.

[25] T. Cao, W. S. Ng, W. L. Goh, and Y. Gao, "DWT-PoolFormer: Discrete wavelet transform-based quantized parallel PoolFormer network implemented in FPGA for wearable ECG monitoring," *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Xi'an, China, 2024, pp. 1-5, doi: 10.1109/BioCAS61083.2024.10798386.

[26] N. Ibtehaz *et al.*, "EDITH: ECG biometrics aided by deep learning for reliable individual authentication," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 4, pp. 928–940, Aug. 2022, doi: 10.1109/TETCI.2021.3131374.

[27] J. Xiao *et al.*, "ULECGNet: An ultra-lightweight end-to-end ECG classification neural network," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 1, pp. 206–217, Jan. 2022, doi: 10.1109/JBHI.2021.3090421.

[28] D. Lee, S. Lee, S. Oh, and D. Park, "Energy-efficient FPGA accelerator with fidelity-controllable sliding-region signal processing unit for abnormal ECG diagnosis on IoT edge devices," *IEEE Access*, vol. 9, pp. 122789–122800, 2021, doi: 10.1109/ACCESS.2021.3109875.

[29] K. Weimann and T. O. F. Conrad, "Federated learning with deep neural networks: A privacy-preserving approach to enhanced ECG classification," *IEEE J. Biomed. Health Inform.*, vol. 28, no. 11, pp. 6931–6943, Nov. 2024, doi: 10.1109/JBHI.2024.3427787.

[30] H. Chu *et al.*, "A neuromorphic processing system with spike-driven SNN processor for wearable ECG classification," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 4, pp. 511–523, Aug. 2022, doi: 10.1109/TBCAS.2022.3189364.

[31] R. Dekimpe and D. Bol, "ECG arrhythmia classification on an ultra-low-power microcontroller," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 3, pp. 456–466, Jun. 2022, doi: 10.1109/TBCAS.2022.3182159.

[32] Y. Umuroglu *et al.*, "Finn: A framework for fast, scalable binarized neural network inference," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays (FPGA)*, 2017.

[33] W. S. Ng, W. L. Goh, and Y. Gao, "High accuracy and low latency mixed precision neural network acceleration for TinyML applications on resource-constrained FPGAs," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Singapore, 2024, pp. 1–5, doi: 10.1109/ISCAS58744.2024.10558440.