LiMO: A Lightweight MambaOut system for end-to-end IoMT ECG Diagnosis with Fully Configurable Quantization Co-Design on MCU

Tiancheng Cao[1], Hongtao Li[1], Hen-Wei Huang[1,2]
[1]School of Electrical and Electronic Engineering, Nanyang Technological University, Republic of Singapore
[2]Lee Kong Chian School of Medicine, Nanyang Technological University, Republic of Singapore

Email: tiancheng.cao@ntu.edu.sg

***Abstract*—Deep learning frameworks such as Mamba have significantly improved inference speed and computational efficiency. However, deploying them on resource-constrained edge devices on the Internet of Medical Things (IoMT) poses challenges due to the demands for real-time processing, secure data handling, and ultra-low power consumption. To address these constraints, this paper introduces LiMO, a lightweight MambaOut architecture that runs entirely on an microcontroller unit (MCU) for real-time electrocardiogram (ECG) diagnosis. By integrating the proposed fully configurable quantization, scale-linear layer fusion, and a hardware-software co-design approach, LiMO substantially reduces computational loading while maintaining medical-grade accuracy. Notably, the entire model consists of just 1.8k parameters. When deployed on the nRF52840 MCU, LiMO occupies only 54.4 KB of Flash and 10.25 KB of RAM, achieves 98.75% accuracy on the MIT-BIH dataset, and consumes as little as 8.08 mJ per heartbeat classification at 3.3 V. These results demonstrate that LiMO successfully bridges the gap between state-of-the-art deep learning methodologies and the stringent requirements of real-time ECG diagnosis in MCU-based IoMT environments, offering a practical and energy-efficient solution for next-generation medical monitoring.**

Keywords—ECG classification, Software-hardware Co-design, Discrete Wavelet Transform, Poolformer

I. INTRODUCTION

In recent years, the advanced deep learning frameworks [1], like the newly proposed Mamba [2], has significantly advanced artificial intelligence, enhancing inference speed and computational efficiency, and achieving superior performance across diverse fields such as natural language processing and time-series analysis [3-5]. However, deploying such architectures on the Internet of Medical Things (IoMT) domain presents significant challenges. The conventional Mamba framework requires substantial parameters and computational power, making it unsuitable for resource-constrained edge devices typically used in healthcare environments. These devices require real-time data processing, patient privacy, data security, and ultra-low power consumption, particularly for tasks like detecting and diagnosing cardiovascular diseases (CVDs) [6-7]. Therefore, the significant advantages of the Mamba architecture make its adaptation crucial for meeting the stringent demands of medical edge computing, highlighting the need for specialized, efficient solutions tailored for healthcare applications.

In recent years, many studies have focused on deploying advanced AI models on resource-constrained edge devices, particularly in the healthcare domain, to enable real-time and secure IoMT [8-10]. Significant progress has been made in deploying Mamba algorithms on sophisticated yet powerful standalone devices; however, the high resource consumption involved has rendered them impractical for edge devices in medical applications, where computational and energy resources are limited [8]. Additionally, existing approaches for ECG diagnosis on edge devices often face challenges in achieving a balance between model accuracy, computational performance, and parameter efficiency, leading to compromises that hinder effective real-world deployment [9]. Moreover, while quantization techniques have been employed to reduce model size and complexity, current methods fail to optimize the trade-off between precision and bit-width for each individual layer [10]. This limitation prevents achieving the optimal compromise between accuracy preservation and resource efficiency, which is critical for practical edge computing applications. Although MambaOut has demonstrated that the Mamba architecture can be effectively simplified for vision classification tasks [11], while the nonlinear operations involved still pose a significant challenge for practical biomedical hardware implementation. These challenges emphasize the need for an approach that integrates a more efficient adaptation of the Mamba framework on MCU, achieves balanced ECG diagnostic performance, and employs adaptive layer-wise configurable quantization strategies, thus making it viable for real-time healthcare monitoring in constrained environments.

In this work, we present a lightweight MambaOut (LiMO) system that incorporates novel optimization techniques to enhance computational efficiency while preserving model accuracy. To achieve an optimal balance between precision and computational load, we introduce a layer-wise configurable quantization approach, applied to both weights and activations during forward computations. Following a hardware-software co-design methodology, we ensure seamless integration between algorithmic innovations and hardware constraints. The LiMO system, occupying only 55,716 bytes of FLASH (about 54.4 KB) and 10,496 bytes of

The work is sponsored by Nanyang Assistant Professorship Start-up Grant (NASUG).

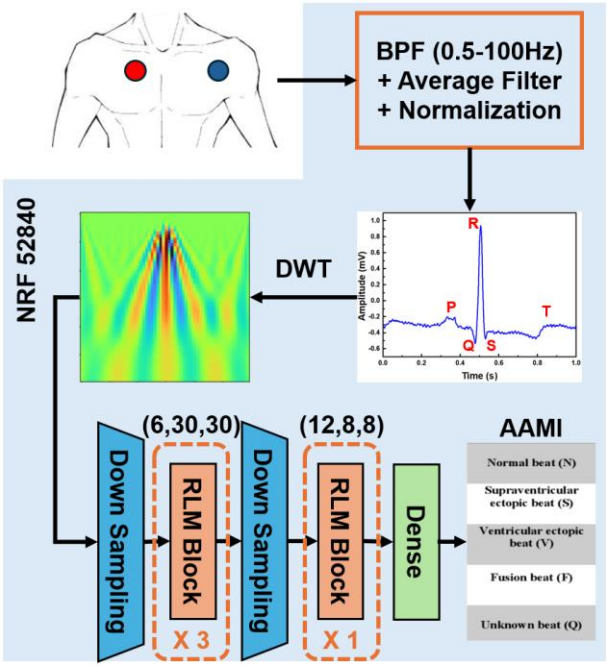

Fig. 1. The architecture of the proposed LiMO model. AAMI is the ECG classification standards developed by the Advancement of Medical Instrumentation.

RAM (about 10.25 KB), was successfully deployed on an nRF52840 MCU (Nordic Semiconductor ASA). Utilizing the MIT-BIH dataset for evaluation, the implementation achieved an impressive accuracy of 98.75%. The key contributions of this study are: (1) balancing model accuracy and efficiency by significantly reducing complexity while maintaining medical-grade prediction accuracy; (2) overcoming MCU hardware limitations, including memory, computational capacity, and power consumption, to enable efficient deep learning deployment; and (3) demonstrating an end-to-end MCU implementation, bridging theoretical advancements with practical applications.

The rest of the paper is organized as follows. Section II provides a comprehensive overview of the proposed LiMO system for edge electrocardiogram classification system. Section III presents the fully configurable quantization strategy to enhance computational efficiency and model compactness. Section IV presents the implementation results on MCU while Section V concludes the paper.

II. LiMO ECG System Software Co-Design

A. LiMO System Overview

The structure of the proposed LiMO system for the end-to-end edge electrocardiogram classification system is depicted in Fig. 1. The single-channel ECG signal is acquired using suction cup electrodes and processed by the nRF52840 MCU. Upon acquisition, the raw signal is first processed with a band-pass filter (BPF, 0.5-100 Hz), followed by a moving average filter and normalization to maintain the amplitude within a standard range, thus mitigating variability arising from inconsistent electrode contact. These preprocessing steps result in a smoother and more distinct ECG signal, facilitating

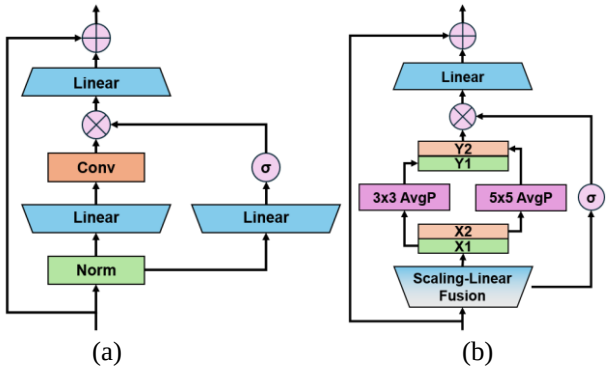

Fig. 2. The block structure of (a) conventional MambaOut block and (b) the proposed Redesigned Lightweight MambaOut (RLM) block.

the identification of key waveform components, including the PR interval, QRS complex, and T wave. Once the waveform is preprocessed, it is segmented into individual samples based on Q-peak detection. Subsequently, each segment undergoes discrete wavelet transform (DWT) for multiresolution analysis [12], generating a wavelet coefficient spectrogram for detailed feature extraction with

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

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

where, $f[n]$ represents the discrete signal at sample n, j and k are the scale and translation of the transform, respectively, and φ is the wavelet function. Finally, the LiMO system is executed on the MCU for real-time diagnosis, providing the predicted classification outcomes.

B. Lightweight MambaOut Block

The conventional block in MambaOut is shown in Fig. 2(a) which is formulated as

$$X' = Norm(X) \quad (2)$$

$$Y = \left(Conv(X'W_1) \odot \sigma(X'W_2)\right)W_3 + X \quad (3)$$

where $Norm(\cdot)$ represents normalization, $Conv(\cdot)$ refers to the module to conduct convolution, $\odot$ is the elementwise multiplication and σ is the active function.

To enhance computational and memory efficiency, the block has been redesigned as the Redesigned Lightweight MambaOut (RLM) Block in this work. Firstly, batch normalization, which introduces inference complexity, has been demonstrated to be replaceable in lightweight models [13]. Therefore, it has been replaced by channel scaling, demonstrated as

$$X' = Linear(Scaling(X)) \quad (4)$$

where the input tensor is X of shape (C, H, W), $Scaling(\cdot)$ is the channel scaling operation with factor of shape $(C, 1, 1)$ and $Linear(\cdot)$ is the linear multiplication with bias.

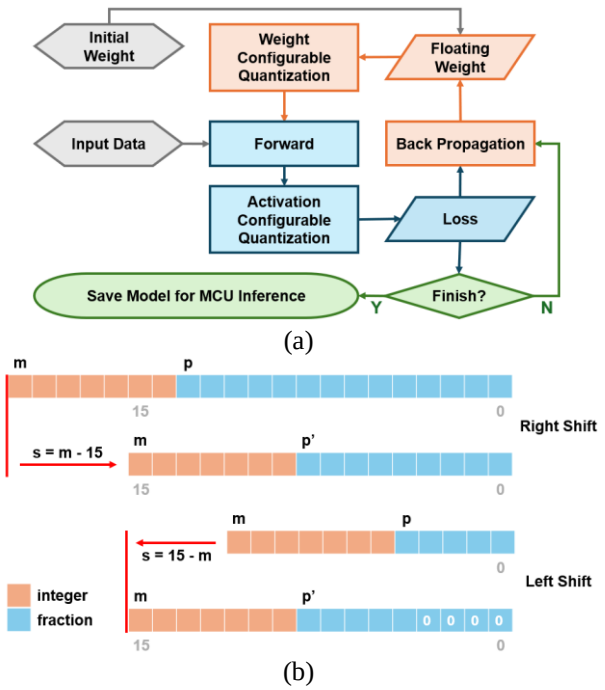

Fig. 3. (a) The proposed fully configurable quantization scheme flow chart. (b) The schematic of configurable shifting operation.

While convolution is computationally intensive due to its high number of multiply-accumulate operations, it is inefficient for deployment on the Internet of Medical Things (IoMT). Considering that channel-wise feature interactions have already occurred in the linear stage, we apply average pooling to extract adjacent features within the same channel [14]. Inspired by the Inception module [15], the feature map is divided into two groups, which undergo 3×3 and 5×5 average pooling, respectively, to capture receptive fields of different resolutions. The approach can be expressed as

$$[X_1;\ X_2] = split(X') \tag{5}$$

$$Y_1 = AvgPool_{3\times3}(X_1), Y_2 = AvgPool_{5\times5}(X_2) \tag{6}$$

$$Y\ =\ cat[Y_1;\ Y_2] \tag{7}$$

where $split(\cdot)$ and $cat(\cdot)$ represents splitting and concatenation along the channel dimension, respectively and $AvgPool_{k\times k}$ is a two-dimensional average pooling operation with a kernel size of k.

Finally, a shortcut path is added and the redesigned block is shown in Fig. 2(b), which significantly reduces the computational burden while maintaining the ability to capture multi-scale features. By replacing batch normalization with channel scaling and utilizing average pooling to enhance feature extraction, the proposed block provides an efficient alternative suitable for resource-constrained environments such as the IoMT.

C. Hyperparameter Optimization

The original MambaOut structure was designed for highly complex datasets, leading to overcapacity when applied to ECG diagnosis. To make the model more lightweight and suitable for this context, key hyperparameters were carefully optimized. Average pooling was applied to reduce the DWT-derived feature maps from (360, 60) to a size of $(60, 60)$, effectively decreasing spatial dimensions while preserving essential information, thus reducing computational burden. A two-stage structure with feature map sizes of $(H/2, W/2)$ and $(H/4, W/4)$ was adopted, using only 6 channels in the first stage and 12 in the second to maintain a compact architecture optimized for memory and computational efficiency.

The modified RLM block incorporates an expansion ratio of 2, controlling intermediate feature growth to balance parameter efficiency with expressiveness. These optimizations result in a model with only 1.8k trainable parameters, making it well-suited for real-time ECG diagnosis on resource-constrained edge devices.

III. Quantization-aware Fusion Hardware Co-design

A. Scaling-Linear Fusion

To ensure efficient operation on an MCU, a hardware co-design approach is adopted, with the channel scaling and linear layers fused in the proposed model as shown in Fig. 2(b). This fusion effectively removes redundant operations, significantly reducing the number of multiply-accumulate (MAC) operations. The derivation of the scaling and linear layer fusion is presented below

$$\begin{aligned} Y_{c,h,w} &= Linear\big(Scaling(X_{c,h,w})\big) \\ &= \sum_{i=1}^{C} W_{c,i} \cdot \big(S_{c,1,1} \cdot X_{c,h,w}\big) + b_c \\ &= \sum_{i=1}^{C} (W_{c,i} \cdot S_{c,1,1}) \cdot X_{c,h,w} + b_c \\ &= Fusion\ Linear\big(X_{c,h,w}\big) \end{aligned} \tag{8}$$

where $W_{c,i}$ represents the weight of the convolution kernel of the c^{th} output channel and the i^{th} input channel, b_c is the bias term and $S_{c,1,1}$ represents the scaling factor for the c^{th} channel.

Such an optimization significantly reduces computational complexity during inference, which is particularly advantageous for resource-constrained IoMT devices requiring efficient computational performance. In addition, the co-design approach to fusing the scaling and linear layers minimizes memory access by enabling direct processing of intermediate feature maps without the need for explicit storage or reloading, resulting in improved data access efficiency.

B. Fully Configurable Quantization

To enhance computational and memory efficiency on IoMT devices, we propose a fully configurable quantization scheme. As illustrated in Fig. 3(a), during both training and

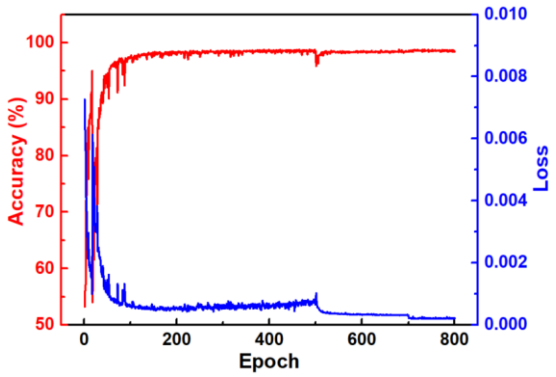

Fig. 4. The proposed LiMO system training accuracy and loss curves, with weight quantization applied at epoch 500 and activation quantization introduced at epoch 700.

inference phases, the weights and activation parameters are quantized to 8-bit and 16-bit, respectively.

Focusing first on activation quantization, conventional integer quantization often struggles to balance the dynamic range of neurons across different layers, resulting in significant precision loss for neurons with small ranges. To address this, we introduce a configurable shifting operation that determines an appropriate fixed-point representation for each layer neurons, thereby achieving both optimal range coverage and minimized precision loss as shown in Fig. 3(b).

Consider a layer of neurons represented in INT32 format with an initial left shift bit p, which is used to represent the fixed-point location. To facilitate efficient quantization, we first convert negative values to positive using two's complement representation. Next, we identify the bit position m of the most significant bit (MSB) of the absolute values among all neurons. All neurons are shifted to adjust the value range accordingly with

$$s = m - 15 \tag{9}$$

If $s > 0$, the neurons are right-shifted by s bits to reduce the value range, whereas if $s < 0$, they are left-shifted by $|s|$ bits to better utilize the available bit range. After the shifting operation, the neurons are truncated to 16 bits, with the original sign restored based on the previously recorded sign bit. The left shift bit p' is then updated as

$$p' = p - s \tag{10}$$

ensuring the accuracy of the fixed-point representation. The final INT16 neurons, along with the updated left shift bit p', serve as the input to the next layer of the neural network, thereby maintaining consistent fixed-point scaling throughout subsequent operations.

For weight quantization, we adopted an approach inspired by dynamic quantization [16], which sets an absolute maximum boundary based on the values within each layer and then rounds them to INT8 levels. Subsequently, a configurable shifting operation is applied to each layer to optimize the utilization of the available bit range, thereby enhancing the efficiency of the quantization process. This scheme ensures a more effective balance of quantization error across layers, leading to improved model performance and stability on resource-constrained IoMT devices.

Fig. 5. Experimental platform based on NRF52840 Development Kit and PPK II and square wave current diagram on laptop

IV. Simulation Results and Discussion

A. Model Training

The training of the proposed LiMO system was performed using an Intel 13600KF CPU alongside an RTX 3060Ti graphics card with GDDR6X memory. Quantization-aware training (QAT) was employed to mitigate the quantization errors that may arise during hardware implementation, as well as to validate the effectiveness of the hardware-software co-design approach [16]. The QAT ensures that the model learns to adapt to the reduced precision during training, leading to improved performance in the final MCU deployment. The results, presented in Fig. 4, demonstrate the LiMO system robustness and efficiency under quantization constraints. At epoch 500, weight quantization was applied, followed by activation quantization at epoch 700. Despite these steps, the system maintained high accuracy and minimal loss, highlighting its ability to optimize computational efficiency without compromising performance.

B. Hardware Deployment And Power Consumption Measurement

Fig. 5 shows the energy consumption of the inference process, with the vertical axis being current and the horizontal axis being time. The voltage is kept constant at 3.3 V. Fig. 6 shows the experimental platform based on the NRF52840 development kit and PPK II, on which the proposed LiMO system has been successfully deployed. The nRF52840 MCU platform integrates 1 MB Flash memory and 256 KB RAM. After optimization, the resource occupancy rates were significantly reduced: the Flash memory consumption reached only 55,716 bytes (5.31% of total capacity), while the peak RAM usage measured 10,496 bytes (4.00% of total capacity). This demonstrates that LiMO achieves efficient memory compression while maintaining model inference accuracy.

To evaluate the system energy efficiency characteristics, an experimental setup employing the Power Profiler Kit II (PPK II, Nordic Semiconductor) was implemented: The PPK II received power via a USB interface and delivered a stabilized 3.3V output voltage to the target development board. Fig. 5 reveals a characteristic square-wave current

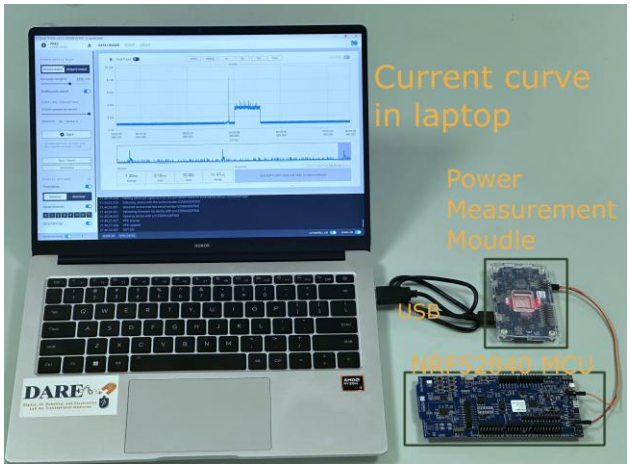

Fig. 6. Experimental platform based on NRF52840 Development Kit and PPK II and square wave current diagram on laptop

profile observed in the software interface, during LiMO activation computations, the operating current surged from a static baseline of 1 mA to over 3 mA. This square-wave power consumption pattern shows close correlation with the temporal characteristics of convolutional layer weight loading and matrix multiplication-accumulation operations.

Further energy consumption analysis was conducted by acquiring time-current raw data through the nRF Connect for Desktop software (sampling rate: 100 kHz). The exported CSV-formatted data underwent numerical integration for periodic energy calculation:

$$E = \int_{t_0}^{t_1} V_{supply} \cdot I(t)dt \qquad (11)$$

The supply voltage was set to V_{supply} = 3.3 V, and numerical integration was implemented using the trapz function in MATLAB. The average energy consumption per inference task was determined to be 8.08±0.23 mJ from 100 independent measurements. These results show that the LiMO architecture maintains medical-grade computational accuracy while meeting the endurance requirements of wearable ECG monitoring devices, which need to perform tens of thousands of inference tasks per day, thus providing a scalable solution for edge intelligence deployment in IoMT devices.

The comparative analysis of the models is succinctly summarized in Table. I. It becomes evident that the proposed LiMO system, refined through a software-hardware co-design approach, exhibits a clear advantage in terms of reduced resource consumption while incurring minimal loss in accuracy. This underscores the efficacy of the proposed model in balancing computational efficiency with performance integrity.

Future work will focus on exploring advanced optimization techniques and developing custom hardware accelerators to further enhance the performance of the LiMO system. Specifically, we plan to integrate additional biosignals such as photoplethysmography (PPG) to improve diagnostic capabilities and provide a more comprehensive analysis. Moreover, we will investigate personalization techniques to tailor the health monitoring system to individual users, thus enhancing accuracy and reliability. Attention will also be given to biometric security, ensuring that the system is not only efficient but also secure against unauthorized access. These efforts aim to push the boundaries of current wearable healthcare technologies, paving the way for next-generation wearable health monitoring systems that are capable of providing more sophisticated, accurate, and instantaneous diagnostics directly from the edge IoMT.

Table. I. PERFORMANCE COMPARISON WITH OTHER APPROACHES

Method	[17]	[18]	[19]	**Proposed**
Feature	CWT	Raw ECG	Raw ECG	CWT
Algorithm	PCA& RNN	ULECGNet	CNN& MLP	Lightweight MambaOut
Test Accuracy(%)	97.7	98.33	97.3	**98.75**
No.of Parameters	46k	8.2k	198k	**1.8k**
Quantization	N.R	N.R	N.R	**16b/8b**
Platform	N.R	MSP432	MSP432	NRF52840
consumption (mJ/s)	N.R	3.1	21.06	**8.08**

N.R.: Not reported

V. CONLUSION

This paper has presented LiMO, a lightweight MambaOut architecture designed to operate entirely on an MCU for real-time ECG diagnosis in IoMT applications. Through the integration of a fully configurable quantization strategy, scale-linear layer fusion, and a hardware-software co-design approach, LiMO demonstrates that advanced deep learning methods can be adapted to severely resource-limited environments. The system achieves 98.75% classification accuracy on the MIT-BIH dataset with just 1.8k parameters, occupying only 54.4 KB of Flash and 10.25 KB of RAM while consuming 8.08 mJ per heartbeat classification at 3.3 V. Experimental results show that LiMO preserves high-precision computation while satisfying the hardware and battery constraints of wearable medical devices, offering a practical architecture for real-time IoMT applications.

REFERENCES

[1] A. Vaswani, et al.,"Attention is all you need," *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[2] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023.

[3] P. Lu *et al.*, "Improving classification of tetanus severity for patients in low-middle income countries wearing ECG sensors by using a CNN-Transformer network," *IEEE Trans. Biomed. Eng.*, vol. 70, no. 4, pp. 1340-1350, April 2023, doi: 10.1109/TBME.2022.3216383.

[4] 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.*, doi: 10.1109/TVLSI.2024.3472270.

[5] X. Xu, C. Chen, Y. Liang, B. Huang, G. Bai, L. Zhao, and K. Shu, "SST: multi-scale hybrid Mamba-Transformer experts for long-short range time series forecasting," *arXiv preprint arXiv:2404.14757*, 2024.

[6] 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," *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Toronto, ON, Canada, 2023, pp. 1-5, doi: 10.1109/BioCAS58349.2023.10389002.

[7] R. Mao *et al*., "An ultra-energy-efficient and high accuracy ECG classification processor with SNN inference assisted by on-chip ANN learning," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 5, pp. 832-841, Oct. 2022, doi: 10.1109/TBCAS.2022.3185720.

[8] S. Li *et al*., "HARMamba: efficient and lightweight wearable sensor human activity recognition based on bidirectional Mamba," *IEEE Internet Things J.*, doi: 10.1109/JIOT.2024.3463405.

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

[10] B. -H. Kung, P. -Y. Hu, C. -C. Huang, C. -C. Lee, C. -Y. Yao and C. -H. Kuan, "An efficient ECG classification system using resource-saving architecture and random forest," *IEEE J. Biomed. Health. Inf.*, vol. 25, no. 6, pp. 1904-1914, June 2021, doi: 10.1109/JBHI.2020.3035191.

[11] W. Yu and X. Wang, "MambaOut: do we really need mamba for vision?" arXiv preprint arXiv:2405.07992, 2024.

[12] 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," 2024 IEEE Biomedical Circuits and Systems Conference (BioCAS), Xi'an, China, 2024.

[13] 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, " *2023 IEEE International Symposium on Circuits and Systems (ISCAS)*, Monterey, CA, USA, 2023, pp. 1-5, doi: 10.1109/ISCAS46773.2023.10181612.

[14] W. Yu, M. Luo, P. Zhou, C. Si, Y. Zhou, X. Wang, J. Feng, and S. Yan, "MetaFormer is actually what you need for vision," *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *arXiv preprint arXiv:1512.00567*, 2015.

[16] 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. Emerging Sel. Top. Circuits Syst.*, vol. 12, no. 4, pp. 934-943, Dec. 2022, doi: 10.1109/JETCAS.2022.3214334.

[17] S. Saadatnejad, M. Oveisi, and M. Hashemi, "LSTM-based ECG classification for continuous monitoring on personal wearable devices," *IEEE J. Biomed. Health Inform.*, vol. 24, no. 2, pp. 515-523, Feb. 2020, doi: 10.1109/JBHI.2019.2911367.

[18] 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.

[19] N. Wang, J. Zhou, G. Dai, J. Huang, and Y. Xie, "Energy-efficient intelligent ECG monitoring for wearable devices," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 5, pp. 1112-1121, Oct. 2019, doi: 10.1109/TBCAS.2019.2930215.