



Lightweight CNN Architectures for Real-time ECG Arrhythmia Classification on Edge Devices

Lokesha HR¹, Poorani S², Prasanna Shete³, Vimalapriya MD⁴,
Sangita Gautam Lade⁵, Shital Parag Dongre⁶, Jegajothi B^{7*}

¹Electronics and Communication, The National Institute of Engineering South, Mananthavady Road, Mysore, Karnataka, India, ²Computer Technology-UG, Kongu Engineering College, Perundurai, Erode, India, ³Computer Engineering, K J Somaiya School of Engineering, Maharashtra, India, ⁴Department of MCA, MEASI Institute of Information Technology, Royapettah, India, ⁵Computer Engineering, Vishwakarma Institute of Technology, Pune, Maharashtra, India, ⁶Artificial Intelligence and Data Science, Vishwakarma Institute of Technology, Pune, India, ⁷SRS Tech Solutions, Chennai, Tamil Nadu, India. *Corresponding Author's Email: jegajothibalakrishnan@gmail.com

Abstract

Real-time and accurate detection of cardiac arrhythmias in electrocardiogram (ECG) signals is essential in clinical decision support and remote health monitoring especially on wearable and edge devices of low power. This paper suggests a BMEL-CNN (Balanced Mixup-Enhanced Lightweight CNN) -based pipeline to classify arrhythmia based on the MIT-BIH Arrhythmia Database. The two significant issues covered by the proposed strategy include the disproportion in classes and underrepresentation of minority-classes. To begin with, there is class balancing and targeted augmentation used to balance the occurrence of N, S, V, F and Q beat types. Second, Mixup augmentation is put forward to enhance generalization by creating artificial soft-labeled samples. Lastly, a categorical focal loss, which is used together with an AdamW optimizer and cosine learning rate schedule, is utilized to emphasize learning in challenging examples and to avoid overfitting. The results of an experiment with standard MIT-BIH test split indicate that the model proposed attains 77.2% overall accuracy, macro-F1-score of 0.41 and 37.3% recall of S-class beats, which is significantly high compared to a baseline CNN model (macro-F1 = 0.37, S-class recall 0%). The network achieves similarly high recall 87.8% on V-class and 93.0% on N-class and the robustness of the network is confirmed. The last model is much smaller in size (around 220 KB) and can take around 4 ms to perform inference on a Raspberry Pi 4, which can be successfully deployed in practical real-time edge monitoring systems.

Keywords: Arrhythmia Classification, BMEL-CNN, Convolutional Neural Network (CNN), Edge Computing, Electrocardiogram (ECG), Mixup Data Augmentation.

Introduction

Cardiovascular diseases are one of the most important causes of death worldwide and cardiac arrhythmias are one of the main causes of cardiovascular morbidity. As a result, early and accurate detection of abnormal heart rhythms is crucial for timely clinical intervention and patient outcomes. Although Electrocardiography (ECG) is the most widely used diagnostic tool for monitoring cardiac activity, manual interpretation of ECG signals is time consuming, labor intensive and prone to human error especially for long-term monitoring of patients, as in wearable healthcare systems and remote patient monitoring. The performance of automated ECG analysis systems has greatly improved in recent years with advances in deep learning. Convolutional neural networks (CNNs) have shown remarkable success, among other approaches, since they are able to

learn discriminative features automatically from raw ECG signals without the need of extensive handcrafted feature engineering. Some studies have shown that deep learning classifiers of ECG can perform as well as an expert cardiologist in the detection of arrhythmias. Moreover, combining deep neural networks with sophisticated signal processing methods has led to the development of end-to-end classification systems that can process intricate ECG patterns and extended ECG segments (up to 10 seconds) (1–3). This disproportion can lead to biased classification models, which in turn can lead to poor detection of clinically important minority classes. To face this problem several studies have tried tackling it by using patient-specific models, data resampling strategies and dedicated training procedures; but minority-class recall is still relatively low in many cases (4).

This is an Open Access article distributed under the terms of the Creative Commons Attribution CC BY license (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution and reproduction in any medium, provided the original work is properly cited.

(Received 05th February 2026; Accepted 25th August 2026; Published 10th October 2026)

The second challenge is related to the computational efficiency and deployment feasibility. Many of the existing architectures are high memory and processing demanding and although they might be high classification accurate, they are computationally expensive. The real-time ECG monitoring requires that the classification model be both accurate and efficient in the calculation. The focus of recent research has thus been on lightweight deep learning architectures and edge-aware implementations that allow for efficient execution on embedded hardware platforms, while maintaining classification performance (5, 6). Besides light architectures, there are advanced training strategies which have been searched to boost the robustness of classification. Focal loss has been proven to improve the minority class recognition by increasing the weight of hard training samples and decreasing the weight of the easily classified samples (7). There are a variety of these methods and Mixup augmentation is one of them that can create synthetic training samples by linearly interpolating two training samples and their labels, which can be used to decrease overfitting and enhance the robustness of the model. Recent research shows that Mixup-based augmentation can lead to a substantial increase in macro-F1 scores and performance on minority classes over traditional oversampling methods (8). These advances have further fueled the research on deployable systems for ECG classification with the advent of edge computing. The feasibility of performing arrhythmia classification using embedded platforms with acceptable latency and accuracy has been shown in TensorFlow Lite implementations (9). Extensive survey on edge-enabled ECG monitoring has highlighted the need for compression and quantization of models and efficient design of architecture for real-time healthcare applications (10). Different studies have proposed attention-based CNN models that enhance the interpretability of the classifiers and the recognition of the minority classes (11) and also, knowledge distillation methods have been used to obtain smaller student networks for wearable devices with low performance loss (12). Moreover, cross-dataset generalization, as a key requirement for the real world, has stimulated the creation of transferable ECG classification frameworks that can preserve their accuracy on different patient sets (13). Recent works have also

investigated temporal convolutional networks and edge-AI acceleration, to achieve computational efficiency without compromising classification performance (14, 15). Furthermore, the Transformer architecture has shown its remarkable ability to model long-range temporal dependencies and extract discriminative features from ECG signals (16). In recent years, research has been conducted to develop resource-efficient deep learning frameworks for classification of ECG arrhythmia, with a particular focus on achieving high accuracy and interpretability and models that explain the decisions made using visual explanations have been proposed in limited computational environments for clinical decision making (17). A compact CNN architecture, incorporating matched-filter concepts, has been proposed to facilitate inter-patient ECG classification and arrhythmia detection for deployment on edge devices that have limited processing power (18). The future of this technology holds promise for even greater innovations, such as the use of deep learning on edge computing devices for the detection of heart arrhythmias from an ECG or the integration of AI with x-rays and CT scans for surgical assistance (19). Model size is around 220KB and inference time is around 4ms per sample, which shows that BMEL-CNN is well suited for resource constrained monitoring applications. This finding is in line with recent evidence to show that convolutional neural networks can be implemented on a device for real-time cardiac abnormality detection (20). More recent studies have investigated the AI powered ECG analysis frameworks that can be deployed on edge devices, targeting the enhancement of the performance of the arrhythmia classification, trainability and computational efficiency (21). The study is motivated by these problems and proposes a Balanced Mixup-Enhanced Lightweight Convolutional Neural Network (BMEL-CNN) for automated ECG arrhythmia classification. The proposed framework includes class balancing, Mixup data augmentation, categorical focal loss, AdamW optimization and cosine learning-rate scheduling to boost minority-class recognition, yet preserve computational efficiency. The model is also quantized and converted to TensorFlow Lite for easy deployment on edge devices.

The main contributions of this work are listed below:

- a) Targeted augmentation is used to create a balanced training set by augmenting the minority heartbeat classes of the MIT-BIH Arrhythmia Database.
- b) A mixup technique is included to improve the generalization of the model and prevent overfitting by generating soft labels synthetic samples.
- c) A sturdy training scheme with categorical focal loss, AdamW optimization and cosine learning-rate scheduling is designed to enhance the stability of learning and minority-class discrimination.
- d) A lightweight CNN architecture is designed and optimized to work on the edge, leading to a small TensorFlow Lite model with low inference latency for deployment in real-time applications in the field of wearable healthcare.
- e) The experimental results show that the proposed BMEL-CNN framework has good minority-class classification performance with low computational overhead. The above features render the proposed approach as a practical solution for real-time arrhythmia monitoring in wearable and portable health care systems.

Methodology

The suggested method seeks to fix class imbalance, make generalization better and let edge devices with limited resources detect arrhythmias in real time. There are four main steps in the overall pipeline shown in Figure 1: (i) preprocessing the data and extracting windows, (ii) balancing the dataset and adding Mixup, (iii) designing and training a lightweight CNN model with focal loss and (iv) deploying the model with TensorFlow Lite. The proposed model's block diagram is shown in Figure 1.

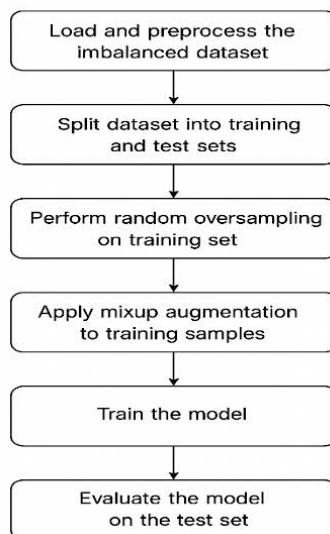


Figure 1: Block Diagram of the Proposed Model

Data Pre-processing

ECG recordings from the MIT-BIH Arrhythmia Database are first loaded and R-peaks are detected using annotation files. Each beat is segmented into a fixed-length window of 200 samples (≈ 0.55 s)

centered around the R-peak to capture sufficient temporal context. Signals are normalized on a per-window basis using z-score normalization to stabilize network training (22). Dataset summary of the proposed model is shown in Table 1.

Table 1: Dataset Summary

Class	Label(s) Included	Train Samples	Test Samples
N (Normal)	N, L, R, e, j	52,665	34,341
S (Supraventricular)	S, A, a, J	973	1,795
V (Ventricular)	V, E	3,787	3,243
F (Fusion)	F	410	1,110
Q (Unknown/Noise)	Q, /, f	13	5,504
Total	–	57,848	45,993

Dataset Balancing and Augmentation

The original dataset is highly imbalanced, with normal beats dominating the training set. To mitigate this, minority-class samples (S, V, F, Q) are oversampled until their counts approach that of the majority class (N). Light augmentations (Gaussian noise, amplitude scaling, random shifts)

are applied to avoid overfitting. Additionally, Mixup augmentation is applied to generate synthetic training pairs, producing soft-labeled combinations of beats that improve generalization and reduce decision boundary overconfidence. Algorithm 1 BMEL-CNN Training Procedure with AdamW is shown below.

Algorithm 1: BMEL-CNN Training Procedure with AdamW

Input: Pre-processed ECG dataset (X_{train} , y_{train})

Output: Trained BMEL-CNN model and TFLite deployment file

(a) Initialize BMEL-CNN model with:

3 × Conv1D layers + BatchNorm + MaxPool
Global Average Pooling + Dropout
Dense Softmax output (5 classes)

(b) Data Preparation:

Apply class balancing with oversampling and light augmentation
Apply Mixup augmentation to generate soft-labeled synthetic pairs

(c) Training Configuration:

Loss function: Categorical Focal Loss ($\gamma = 2.0$)
Optimizer: AdamW (learning rate = $3e-3$, weight decay = $1e-4$)
Learning rate schedule: Cosine Decay Restarts
Batch size = 256, epochs = 30
EarlyStopping (patience = 6) and Model Checkpoint enabled

(d) Training Loop:

for epoch = 1 to max_epochs do
 Forward pass: Compute predictions for each batch
 Compute loss using focal loss
 Backpropagate gradients and update weights using AdamW
 Monitor validation accuracy for checkpointing
 if validation accuracy does not improve for 'patience' epochs then stop training

(e) Evaluation:

Compute accuracy, precision, recall, macro-F1 on test set
Generate confusion matrix and ROC curves per class

(f) Deployment:

Quantize model and convert to TensorFlow Lite
Deploy on edge device (e.g., Raspberry Pi) and benchmark inference time

Model Architecture

An adaptive BMEL-CNN has three 1-D convolutional layers, batch normalization and max-pooling processes that work together to gradually pull-out temporal characteristics while lowering the number of dimensions. A global average pooling layer gets the global context and then dropout ($p=0.5$) is used to regularize the data. A completely connected softmax layer gives the chances of each of the five beat classes (N, S, V, F and Q). The model has fewer than 250,000 parameters, which makes it good for use in embedded systems. The model is trained

with categorical focus loss ($\gamma=2.0$), which puts more weight on samples that are hard to identify, especially those from minority classes. The AdamW optimizer with cosine learning rate scheduling makes sure that convergence is smooth and that overfitting doesn't happen. To keep the optimum validation performance, early halting and model checkpointing are used. A batch size of 256 is used for training. The trained model is quantized and changed to TensorFlow Lite format for real-time monitoring apps. This makes it take up about 220 KB of RAM. The model takes about 4 ms to make an inference for each window on a Raspberry Pi 4. This means that arrhythmia can be

classified almost in real time on edge devices. The suggested method uses balanced data creation, Mixup augmentation, focal loss optimization and a compact CNN design to find a compromise between accuracy and efficiency. Compared to a baseline CNN, this method enhances S-class recall by more than 35% while keeping strong N and V detection performance. This makes it good for

clinical and wearable ECG monitoring situations (23).

Mathematical Analysis of BMEL-CNN

This section presents the mathematical formulation of the proposed Balanced Mixup-Enhanced Lightweight Convolutional Neural Network (BMEL-CNN) framework for ECG arrhythmia classification.

Problem statement is shown in Equation [1],

Let,

$$D = \{(x_i, y_i)\}_{i=1}^N \quad [1]$$

D , denote the training set represents the class of the corresponding heartbeat. ECG segments are first normalized using z-score normalization to address the amplitude variations of the ECG signals and to make training more stable before feature extraction and classification. The normalization process and calculation of the statistical measures are defined as follows,

Where,

$x_i \in \mathbb{R}^L$ is a 1D segment of length L samples (here $L = 200$) and

$y_i \in \{1, \dots, C\}$ is the class label over $C = 5$

Heartbeat types,

The five categories of the heartbeat signal that are studied in this work are those which are described by the annotation protocol of the MIT-BIH Arrhythmia Database: Normal (N), Supraventricular ectopic (S), Ventricular ectopic (V), Fusion (F) and Unknown (Q).

We use the one-/hot encoding $e_{y_i} \in \{0,1\}^C$ for label y_i .

Each window is z-/score normalized and shown as Equation [2].

$$\tilde{x}_i = \frac{x_i - \mu(x_i)}{\sigma(x_i) + \varepsilon}, \mu(x_i) = \frac{1}{L} \sum_{t=1}^L x_{i,t}, \sigma(x_i) = \sqrt{\frac{1}{L} \sum_{t=1}^L (x_{i,t} - \mu)^2} \quad [2]$$

with a small $\varepsilon > 0$ for numerical stability.

Balanced Sampling and Oversampling

Let n_c be the number of training samples in class c .

BMEL-/CNN performs class balancing by matching each minority class count to $n_{\max} = \max_c n_c$ via random oversampling (with light perturbations).

Formally, the balanced set D_{bal} is obtained by drawing is shown in Equation [3].

$$D_{\text{bal}} = \bigcup_{c=1}^C \{(A(\tilde{x}), c) \mid \tilde{x} \sim D_c, \text{repeat until count } n_{\max}\} \quad [3]$$

where $A(\cdot)$ applies small additive noise, amplitude scaling and circular shifts to avoid duplicate leakage.

Mixup With Soft Labels

To further regularize the decision boundary, we apply Mixup on balanced batches.

Given two samples $(\tilde{x}_i, e_{y_i})$ and $(\tilde{x}_j, e_{y_j})$ and $\lambda \sim \text{Beta}(\alpha, \alpha)$, $\alpha > 0$, the mixed pair is shown in Equations [4, 5].

$$x^{(\text{mix})} = \lambda \tilde{x}_i + (1 - \lambda) \tilde{x}_j, \quad [4]$$

$$y^{(\text{mix})} = \lambda e_{y_i} + (1 - \lambda) e_{y_j} \quad [5]$$

The soft label $y^{(\text{mix})} \in \Delta^{C-1}$ (probability simplex) encourages smooth posteriors and reduces overconfidence.

Lightweight CNN Forward Model

BMEL-CNN uses three 1D convolutional blocks with batch normalization and pooling, followed by Global Average Pooling (GAP) and a fully connected softmax head is modelled in Equation [6].

Let $h^{(0)} = x^{(\text{mix})}$ denote the input.

For block $b \in \{1,2,3\}$:

$$z^{(b)} = h^{(b-1)} * W^{(b)} + b^{(b)}, u^{(b)} = \phi \left(\text{BN} \left(z^{(b)} \right) \right), h^{(b)} = \text{MaxPool}(u^{(b)}) \quad [6]$$

where $*$ is 1D convolution, $\phi(\cdot)$ is ReLU and BN is batch normalization.

GAP reduces the temporal dimension is shown in Equation [7]:

$$g = \text{GAP}(h^{(3)}) \in \mathbb{R}^K \quad [7]$$

optionally followed by dropout q is modeled in Equation [8]:

$$\tilde{g} = \text{Dropout}_q(g) \quad [8]$$

The class logits and posteriors are shown in Equation [9],

$$o = W_o^T \tilde{g} + b_o, p^{\wedge}_c = \text{softmax}(o), p^{\wedge}_c = \frac{\exp(o_c)}{\sum_{k=1}^C \exp(o_k)} \quad [9]$$

Categorical Focal Loss with Soft Labels

To emphasize hard examples, we use the categorical focal loss applied to soft (Mixup) targets are in Equations [10, 11]:

$$L_{\text{CE}}(y, p^{\wedge}) = - \sum_{c=1}^C y_c \log p^{\wedge}_c, \quad [10]$$

$$L_{\text{focal}}(y, p^{\wedge}; \gamma) = - \sum_{c=1}^C (1 - p^{\wedge}_c)^{\gamma} y_c \log p^{\wedge}_c, \gamma \geq 0 \quad [11]$$

With L2 weight decay applied to convolution and dense kernels, the total loss is shown in Equation [12].

$$J = \frac{1}{B} \sum_{i=1}^B L_{\text{focal}}(y^{(i)}, p^{\wedge(i)}; \gamma) + \lambda_{l_2} \sum_{\theta \in W} \|\theta\|_2^2 \quad [12]$$

Where, B is batch size, W is the set of trainable weights subject to decay and $\lambda_{l_2} > 0$.

For a target component $y_c > 0$, the multiplicative term $(1 - p^{\wedge}_c)^{\gamma}$ increases the effective gradient when $p^{\wedge}_c$ is small (hard examples) and shrinks it when $p^{\wedge}_c$ is large (easy examples).

Cosine Decay Restarts and AdamW

Let, t be the global step and T_0 the first cycle length (in steps).

The cosine decay with restarts learning rate is shown in Equation [13],

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min}) \left[1 + \cos \left(\pi \frac{t - t_r}{T_r} \right) \right] \quad [13]$$

for $t \in [t_r, t_r + T_r)$ within restart r , where $T_{r+1} = T_r \cdot t_{\text{mul}}$ and $t_{r+1} = t_r + T_r$

We optimize with AdamW (decoupled weight decay) is shown in Equations [14-17]:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla_w J_t, \quad [14]$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\nabla_w J_t)^{\odot 2} \quad [15]$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \hat{v}_t = \frac{v_t}{1 - \beta_2^t}, \quad [16]$$

$$w_{t+1} = w_t - \eta_t \left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} + \lambda_w w_t \right) \quad [17]$$

Where, λ_w is the decoupled decay coefficient (weight decay) and $\epsilon > 0$ ensures stability.

Early Stopping

Let, ValAcc(e) be validation accuracy at epoch

e. Training stops at epoch e^* if no improvement is seen for P consecutive epochs are modelled in Equation [18]:

$$e^* = \arg \max_{e \in E} \text{ValAcc}(e), \text{ stop if } \text{ValAcc}(e - k) \geq \text{ValAcc}(e) \quad \forall k \in \{1, \dots, P\} \quad [18]$$

Evaluation Metrics

From the confusion matrix $M \in \mathbb{N}^{C \times C}$ with M_{ij} the count of true class i predicted as j , modelled in Equations [19, 20]:

$$TP_c = M_{cc}, FN_c = \sum_{j \neq c} M_{cj}, FP_c = \sum_{i \neq c} M_{ic}, TN_c = \sum_{i \neq c} \sum_{j \neq c} M_{ij}, \quad [19]$$

$$\text{Prec}_c = \frac{TP_c}{TP_c + FP_c + \epsilon}, \text{Rec}_c = \frac{TP_c}{TP_c + FN_c + \epsilon}, F1_c = \frac{2 \text{Prec}_c \text{Rec}_c}{\text{Prec}_c + \text{Rec}_c + \epsilon}. \quad [20]$$

Macro averages emphasize balance across classes are shown in Equation [21]:

$$\begin{aligned} \text{Macro} - \text{Prec} &= \frac{1}{C} \sum_{c=1}^C \text{Prec}_c, \text{Macro} - \text{Rec} \\ &= \frac{1}{C} \sum_{c=1}^C \text{Rec}_c, \text{Macro} - F1 = \frac{1}{C} \sum_{c=1}^C F1_c \end{aligned} \quad [21]$$

Overall accuracy is shown in Equation [22],

$$\text{Acc} = \frac{\sum_{c=1}^C M_{cc}}{\sum_{i=1}^C \sum_{j=1}^C M_{ij}}. \quad [22]$$

ROC Curves and AUC

For class c , define scores $s^{(i)} = p_c^{(i)}$ and binary ground truth $z^{(i)} = 1[y^{(i)} = c]$

Varying a threshold $\tau \in [0, 1]$ yields is shown in Equation [23],

$$\text{TPR}_c(\tau) = \frac{\sum_i 1[s^{(i)} \geq \tau] \cdot z^{(i)}}{\sum_i z^{(i)}}, \text{FPR}_c(\tau) = \frac{\sum_i 1[s^{(i)} \geq \tau] \cdot (1 - z^{(i)})}{\sum_i (1 - z^{(i)})} \quad [23]$$

The area under the ROC curve is shown in Equation [24],

$$\text{AUC}_c = \int_0^1 \text{TPR}_c(\text{FPR}_c^{-1}(u)) du \approx \sum_k \text{TPR}_c^{(k)} \Delta \text{FPR}_c^{(k)} \quad [24]$$

Post-training Quantization (Uniform Affine)

To enable edge deployment, we apply 8-bit uniform affine quantization to weights/activations. Given a real tensor $x \in [\min, \max]$, define scale and zero-/point is shown in Equation [25]:

$$s = \frac{\max - \min}{255}, z = \left\lfloor -\frac{\min}{s} \right\rfloor \quad [25]$$

and quantize/dequantize as in Equation [26],

$$q = \text{clip} \left(\left\lfloor \frac{x}{s} + z \right\rfloor, 0, 255 \right), \hat{x} = s(q - z) \quad [26]$$

This yields compact models with negligible accuracy loss but large latency and memory savings on low-power hardware.

In the mathematical formulation of BMEL-CNN, the end-to-end description of the proposed workflow is presented. Z-score transformation is used to normalize the ECG windows in order to stabilize training and class balancing is used to overcome the skewed data distribution. Mixup augmentation formulates soft-label synthetic examples, which promote more smooth decision boundaries and minimizes overfitting. The CNN is lightweight and mathematically, it is a series of convolution, pooling, batch normalization and global average pooling layers with a softmax classifier at the end, which gives the probability of the classes. The training procedure is directed by a categorical focal loss purpose that re-weights difficult to categorize instances coupled with L2 weight regularization to enhance generalization. AdamW with a cosine learning rate restart schedule is used to optimize model parameters to allow the smooth convergence. The standard measures used to evaluate the performance are accuracy, precision, recall, macro-F1 and ROC-AUC based on the confusion matrix. Lastly, the quantization of the model after training is used to decrease the size of the model and make it able to be deployed to resource constrained edge devices (24). This high-quality mathematical support proves that BMEL-CNN is a valid and effective method to deal with real-time arrhythmia classification.

Results

Experimental Setup

The experiments conducted with the BMEL-CNN model is carried out on an IntelCorei7-based workstation with 16 GB of random memory and

NVIDIA GeForce GTX1650 graphics card with the Ubuntu 22.04 LTS operating system. The ECG signals were sampled at 360 Hz and were taken out of the MIT-BIH Arrhythmia Database and windowed into 200 samples with an annotated R-peaks as the center of the window. A standard inter-patient protocol was used to split data between training and testing (22 records each) to make patient-independent evaluation. The MIT-BIH Arrhythmia Database was used to conduct experiments based on the standard inter-patient split: 22 records were used to train and 22 were used to test. The beats were further divided into 200-sample windows, with the R-peak at their center and z-score-normalized. The suggested BMEL- CNN was written on the basis of the TensorFlow 2.x version and trained with a batch size of 256, the AdamW optimizer with restart of cosine learning rates up to 30 epochs. Patience of 6 epochs was used to avoid overfitting. The proposed Balanced Mixup-Enhanced Lightweight Convolutional Neural Network (BMEL-CNN) was tested against a baseline CNN trained using the imbalanced dataset without class balancing as shown in Table 2. The overall accuracy of the baseline CNN was 80.0%, while that of BMEL-CNN was 77.2%. This is a drop of 2.8 points but overall accuracy is not a good indicator of performance for this seriously imbalanced database since it is highly susceptible to correctly classified normal (N) beats. The score of macro-F1 on the other hand grew from 0.37 to 0.41, indicating that all classes are given equal importance. This is a 10.8% relative improvement and 0.04 absolute improvement from the baseline. Overall, the higher macro-F1 score for BMEL-CNN suggests that the model demonstrated a more balanced performance across the five heartbeat classes.

Table 2: Performance Comparison

Metric	Baseline CNN (No Balancing)	Proposed Method (Balancing + Mixup + Focal Loss)
Overall Accuracy	0.80	0.77 (balanced focus)
Macro F1-Score	0.37	0.41
Recall (N)	0.97	0.93
Recall (S)	~0.00	0.37
Recall (V)	0.84	0.88
Recall (F)	0.07	0.04 (still challenging)
Recall (Q)	0.00	0.00

Model Size (TFLite)	-	~220 KB
Edge Inference Speed	-	~4 ms/sample (Raspberry Pi 4, single-thread)

The biggest gain was for the supraventricular (S) class. BMEL-CNN improved its recall rate from ~0% in the baseline model to 37.3%. Therefore, the proposed framework identified the supraventricular beats that were entirely missed by the baseline CNN, correctly identifying over 33%. The enhanced recognition of the clinically relevant S class did not affect the detection of the clinically relevant V (ventricular) class as shown by ventricular-beat recall, which improved from about 84.2% to 87.8%. Normal-beat recall was however somewhat reduced, dropping from about 97% to 93%. This decrease is an indication of the change in the decision boundary, which will be more balanced compared to the majority class. Although F and Q (noise/unknown) beats were still recognized, this was limited. There are relatively a smaller number of representative training examples in these classes and may show morphological similarities with other heartbeat classes. Specifically, there are 13 training samples and 5,504 test samples in the Q class. As a result,

the model does not have enough representative information to learn a generalizable Q-class representation. This result shows how loss reweighting and data augmentation cannot replace an extreme lack of training observations. More advanced class-conditional data generation methods or additional clinically validated recordings may thus be needed to enhance the recognition of the F and Q classes. The behaviour of BMEL-CNN is also explained using the confusion matrix shown in Figure 2 based on the class. The model correctly identified 31,947 N beats, 669 S beats, 2,848 V beats, 52 F beats and one Q beat. Residual majority class bias is seen with most S, F and Q misclassifications being assigned to the dominant class, N. However, from the results presented in Table 2, it was observed that the overall recall of S and V classes was improved when using the three classifiers together (class balancing, Mixup augmentation and focal loss) than using the baseline CNN model.

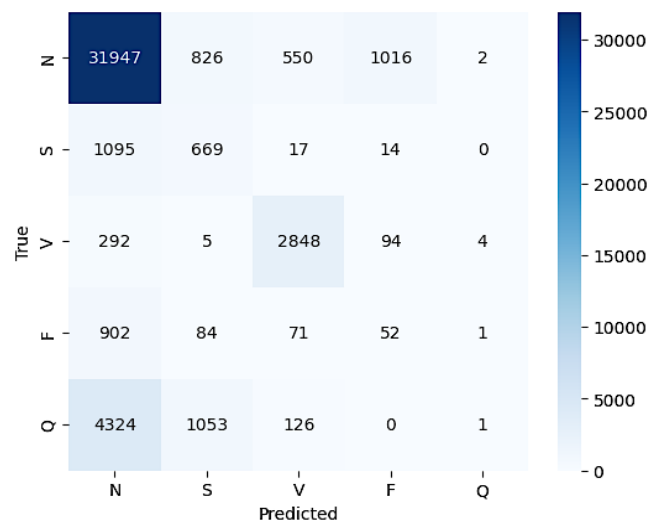


Figure 2: Confusion Matrix for BMEL-CNN Predictions on the MIT-BIH Test Set

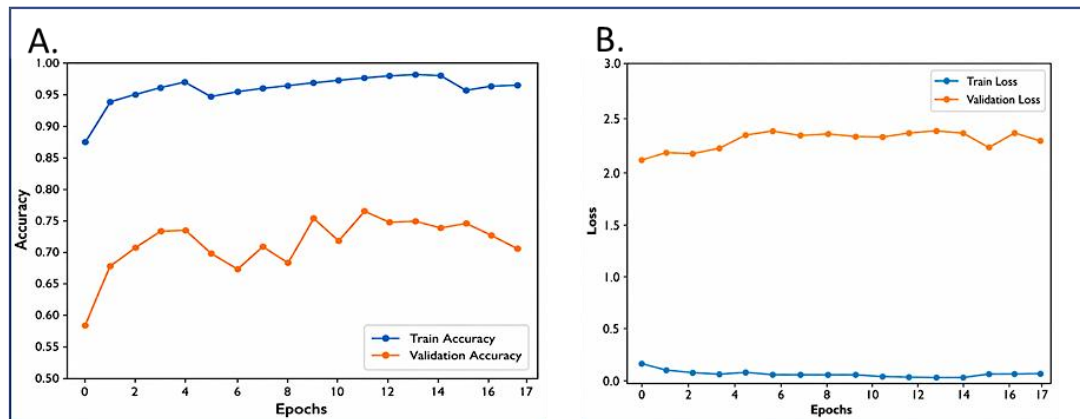


Figure 3: Training and Validation Performance of BMEL-CNN. (A) Accuracy Curves Showing Convergence of Training and Validation Accuracy, (B) Loss Curves Indicating Smooth Optimization and Limited Overfitting Due to Mixup Augmentation and Focal Loss

The findings indicate that the balanced oversampling and Mixup augmentation enhances the representation of minority-class and lowers the bias on model to normal-beats. Categorical focal loss enabled BMEL-CNN to move more focus on samples that are hard to classify, which also enhanced sensitivity on underrepresented classes. It is important to note despite a slight decrease in

overall accuracy relative to the baseline (which is biased toward N-class), the macro-F1 score and per-class recall change is essential to clinical usefulness, in which missing arrhythmic events is more expensive than a limited false positive. Figure 3 illustrates the training and validation Accuracy and Loss curve (25).

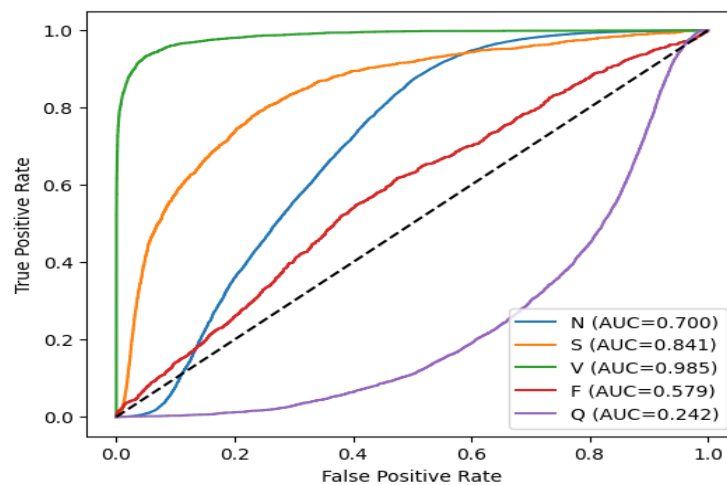


Figure 4: ROC Curves for Each Class Generated Using BMEL-CNN Predictions

Figure 4 shows the ROC curves for each class generated using BMEL-CNN predictions. High area under the curve (AUC) values for N and V classes demonstrate strong separability, while the S-class ROC curve shows significant improvement compared to the baseline, validating BMEL-CNN's effectiveness for minority-class detection. From a deployment perspective, the BMEL-CNN model

remains lightweight, with a quantized TensorFlow Lite model size of ~220 KB and an inference latency of ~4 ms per beat on Raspberry Pi 4. This makes it suitable for real-time monitoring in wearable and edge-based systems. Per-class recall comparison for baseline CNN and BMEL-CNN is shown in Figure 5.

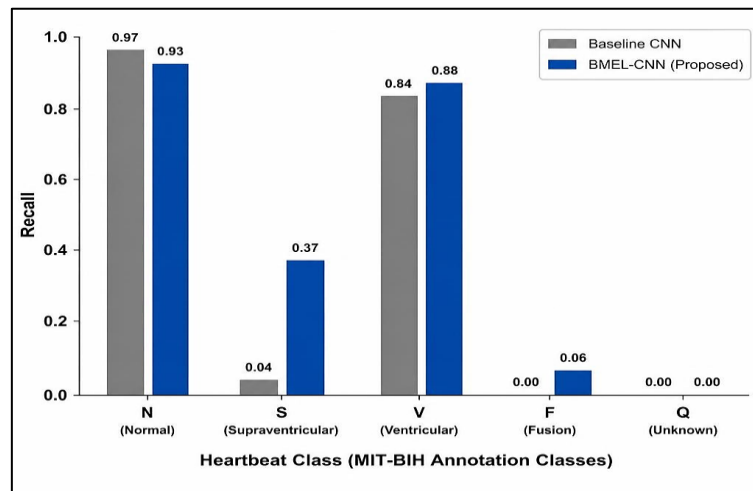


Figure 5: Per-class Recall Comparison for Baseline CNN and BMEL-CNN

Discussion

The experimental results show that the proposed BMEL-CNN framework improves the classification results more than the CNN baseline framework. The overall accuracy decreased from 80.0% to 77.2%, but the macro-F1 score improved from 0.37 to 0.41, which is 10.8% of the difference. The improvement is quite significant for the highly imbalanced MIT-BIH dataset since the overall accuracy is highly dependent on the N class and macro-F1 gives equal weight to all the heartbeat classes. BMEL-CNN's biggest comparative advantage was in the recognition of minorities. The S-class recall rose from just about zero to 37.3%, with a 37.3 percentage point improvement compared to the baseline. Likewise, the percentage of V class recall rose from 84.0% to 87.8%. The results demonstrate the effectiveness of targeted class balancing, Mixup augmentation and categorical focal loss in guiding the learning process towards challenging and underrepresented heartbeat samples. While N-class recall dropped slightly (97.0% to 93.0%), the model maintained excellent normal-beat recognition and significantly improved supraventricular and ventricular arrhythmia detection. The results acquired are consistent with the published findings where the unbalanced sample generation and optimized representation learning leads to better recognition of minority ECG classes (24). In the current scheme, we merge three complementary schemes together, which are minority sample increase, soft-label interpolation (Mixup) and focal loss. In this work, we propose a new scheme that merges three complementary schemes: minority sample increase, soft-label

interpolation (Mixup) and focal loss. The resulting overall effect is shown in the better macro-F1 score and the significant improvement in S-class recall. Normalized RR-interval features and a specialized convolutional initialization has been shown to yield good classification performance in previous inter-patient ECG studies (18). Contrastively, BMEL-CNN significantly outperforms the other methods in terms of minority class improvements with only a simple normalization of 200-sample windows of heartbeats, without relying on handcrafted RR-interval features. This feature helps to maintain a relatively simple end-to-end processing pipeline while minimizing the need for any extra rhythm feature engineering. The results thus confirm the effectiveness of using imbalance-aware training as a method to boost the sensitivity of a lightweight CNN even without explicit RR-interval information. The original representation of the F and Q categories was very sparse, making them difficult categories. During training, there were 13 samples in Q class and 5,504 samples in Q class in the test set. In spite of this large distribution gap, the proposed model achieved not only good macro-F1 rates, but significantly improved S-class recognition. This outcome corroborates the efficacy of the proposed training scheme and illustrates that synthetic augmentation is not a viable substitute for real-world observations at patient-level, especially when the number of real-world observations is very limited. The training curves also indicate that BMEL-CNN was quickly able to learn the expanded heartbeat representations. The accuracy of the training improved from ~87.5% to ~98% and the

accuracy of the validation improved up to ~76.5%. The fluctuations in the following validation show that the model maintained some sensitivity to differences in patients. Still, the use of Mixup, focal loss, cosine learning-rate scheduling and early stopping ensured stable optimization and yielded significant gains in the independent test-set macro-F1 and minority-class recall. Besides the classification results, BMEL-CNN showed that it is capable of being deployed at the edge. The quantized TensorFlow Lite model was about 220 KB and classified one heartbeat in about 4 ms on a Raspberry Pi 4 on a single CPU thread. These values are within the time frame of beat-by-beat ECG monitoring. Current edge-oriented studies of the ECG have focused on the need for low inference latency, compact storage and local processing of signals for wearable and remote-monitoring applications (18, 19, 21). BMEL-CNN meets these requirements, but also includes specific mechanisms for minority-class learning. BMEL-CNN offers an integrated balance between the minority class sensitivity and real-time feasibility on the edge compared with the methods mainly based on classification performance or computational efficiency (18, 19, 21, 24). It has made its main contribution through its 10.8% relative improvement in macro-F1 score, the 37.3 percentage point improvement in the S-class recall score, the 3.8 percentage point improvement in the V-class recall score and its small 220 KB deployment footprint. These results provide a viable base for BMEL-CNN in resource constrained arrhythmia monitoring, where further improvements can be made with the addition of RR-interval information, further F- and Q-class recordings, class-specific threshold calibration and external dataset validation.

Conclusion

The present research proposed BMEL-CNN, an improved lightweight CNN architecture of Mixup with a balanced mode for real-time ECG arrhythmia classification. The model uses a tiny CNN based on edge deployment and a mix of class balancing, Mixup data augmentation and categorical focus loss. The MIT-BIH Arrhythmia Database shows that BMEL-CNN with 77.2% overall accuracy, 0.41 macro-F1 and 37.3% recall of S-class beats beats out a control CNN in terms of minority-class performance and has exceptional N-

and V-class recall. The quantized model is only around 220 KB in size and it takes about 4 ms each beat to run on a Raspberry Pi 4. This means that it can be utilized in real-time and low-power versions of arrhythmia monitoring. These results show that accuracy alone is not enough for clinical significance. BMEL-CNN improves the balance of classes and lowers the number of missed arrhythmic events, which is very important for early diagnosis or patient safety. Future research will focus on improving minority-class recognition with advanced data synthesis techniques, such as GANs or diffusion models, to generate synthesized beats that convincingly resemble authentic sounds. We will also try out CNN-Transformer hybrid models that might be better at modeling temporal features. The research will also try cross-dataset validation on bigger databases (like PTB-XL and INCART) to make sure the results are generalizable. Finally, this study will use explainability methods (like saliency maps) to help doctors understand the results. In future, will also look into hardware-optimized methods like INT8 quantization and pruning to cut down on latency and power use. These methods will work even better on wearable edge devices. The response to these instructions will allow future versions of BMEL-CNN to find unusual arrhythmias more often, work with a wider range of ECGs and do all of this while using less computing power. This will also make it possible to use BMEL-CNN in real-world systems that monitor continuously. Although promising results have been obtained with the proposed BMEL-CNN framework, there are still some limitations. The model was evaluated on the MIT-BIH Arrhythmia Database only and its extension to other ECG databases and different patients is not investigated. Also, the TensorFlow Lite implementation shows low memory usage and low inference latency on a Raspberry Pi 4 and evaluation on a larger number of embedded and wearable devices operating in real-world environments has not been done. Further work is planned to include multi-dataset validation, patient independent evaluation, model quantization and pruning for additional computational optimisation and clinical validation with real time ECG streams from wearable healthcare devices.

Abbreviations

AdamW: Adaptive Moment Estimation with Weight Decay, AI: Artificial Intelligence, CNN: Convolutional Neural Network, Edge AI: Edge Artificial Intelligence, MIT-BIH: MIT-BIH Arrhythmia Database, R-peak: Peak point in ECG signal, TFLite: TensorFlow Lite.

Acknowledgement

We would like to acknowledge the contributions of all those who supported this research. Our sincere thanks go to the participants who took part in the study, as well as to all individuals and institutions whose support made this research possible.

Author Contributions

Lokesha HR, Poorani S and Sangita Gautam Lade: Conceptualization, Conducted the Study, Data Collection; MD Vimalapriya, Shital Parag Dongre and B Jegajothi: Analysis, Interpretation of the Data, Drafting of Manuscript, Review and Editing, Lokesha HR Poorani S, Prasanna Shete, Sangita Gautam Lade: Result Analysis, Supervision, Project Administration. All authors critically reviewed the final manuscript.

Conflict of Interest

The authors have expressed no conflict of interest.

Data Availability

Data is available in the following link:
<https://physionet.org/content/mitdb/1.0.0/>

Declaration of Artificial Intelligence (AI) Assistance

During the preparation of this manuscript, the authors used artificial intelligence (AI)-assisted tools solely to improve the clarity of language, grammar, readability and manuscript formatting. The AI tools were not used to generate, analyze, or interpret the research data, conduct experiments, or formulate the scientific conclusions. All research design, methodology, data analysis, validation and final manuscript content were developed, verified and approved by the authors, who accept full responsibility for the accuracy and integrity of the work.

Ethics Approval

Not applicable.

Funding

The authors declare that no funds, grants, or other

financial support were received during the preparation of this manuscript.

References

1. Gracy GA, Pravin SC. A lightweight residual dilated temporal transformer block for ECG classification on edge devices. *Scientific Reports*. 2026;16:8834. <https://doi.org/10.1038/s41598-026-35531-4>
2. Chen J, Zhang X, Xu L, de Albuquerque VH, Wu W. Implementing the confidence constraint cloud-edge collaborative computing strategy for ultra-efficient arrhythmia monitoring. *Applied Soft Computing*. 2024;154:111402. <https://doi.org/10.1016/j.asoc.2024.111402>
3. Liu Z, Ling X, Zhu Y, Wang N. FPGA-based 1D-CNN accelerator for real-time arrhythmia classification. *Journal of Real-Time Image Processing*. 2025;22:66. <https://doi.org/10.1007/s11554-025-01642-w>
4. Kiranyaz S, Ince T, Gabbouj M. Real-time patient-specific ECG classification by 1-D convolutional neural networks. *IEEE transactions on biomedical engineering*. 2015;63(3):664-75. doi: 10.1109/TBME.2015.2468589
5. Alshuhail A, Alshahrani A, Mahgoub H, Ghaleb M, Darem AA, Aljehane NO, Alotaibi M, Alzahrani F. Machine edge-aware IoT framework for real-time health monitoring: Sensor fusion and AI-driven emergency response in decentralized networks. *Alexandria Engineering Journal*. 2025;129:1349-61. <https://doi.org/10.1016/j.aej.2025.08.030>
6. Aloufi BO, Alhakami W. A lightweight authentication mac protocol for cr-wsns. *Sensors*. 2023;23(4):2015. <https://doi.org/10.3390/s23042015>
7. Ashraf H, Shafiq U, Sajjad Q, Waris A, Gilani O, Boutaayamou M, Bröls O. Variational mode decomposition for surface and intramuscular EMG signal denoising. *Biomedical Signal Processing and Control*. 2023;82:104560. <https://doi.org/10.1016/j.bspc.2022.104560>
8. Liu Y, Wei LF, Qian X, Zhang TH, Chen SL, Yin XC. M3TTS: Multi-modal text-to-speech of multi-scale style control for dubbing. *Pattern recognition letters*. 2024;179:158-64. <https://doi.org/10.1016/j.patrec.2024.02.005>
9. Amara K, Guerroudji MA, Kerdjidi O, Zenati N, Atalla S, Ramzan N. Enhancing arrhythmia diagnosis through ecg deep learning classification deploying and augmented reality 3d heart visualisation and interaction. *IEEE Access*. 2025. doi: 10.1109/ACCESS.2025.3576243.
10. Scrugli MA, Leone G, Busia P, Raffo L, Meloni P. All-Spiking ECG Analysis for Arrhythmia Classification on Low-Power FPGA. *IEEE Sensors Journal*. 2026;26(7):11372-86. doi: 10.1109/JSEN.2026.3663715
11. Vijayan VPH, Rahiman M, Abraham L, Preenu C, MD G. Class-Imbalance-Aware Gated Temporal Attentive Network for ECG Arrhythmia Classification. <https://dx.doi.org/10.2139/ssrn.6029856>
12. Zhang J, Liu A, Gao M, Chen X, Zhang X, Chen X. ECG-based multi-class arrhythmia detection using spatio-temporal attention-based convolutional recurrent neural network. *Artificial Intelligence in Medicine*. 2020;106:101856. <https://doi.org/10.1016/j.artmed.2020.101856>

13. Tao Y, Li Z, Gu C, Jiang B, Zhang Y. ECG-based expert-knowledge attention network to tachyarrhythmia recognition. *Biomedical Signal Processing and Control*. 2022;76:103649.
<https://doi.org/10.1016/j.bspc.2022.103649>
14. Singh S, Singh H, Bueno G, Deniz O, Singh S, Monga H, Hrisheekesha PN, Pedraza A. A review of image fusion: Methods, applications and performance metrics. *Digital Signal Processing*. 2023;137:104020.
<https://doi.org/10.1016/j.dsp.2023.104020>
15. Dong R, Xie L. A multimodal cross-attention-based CNN-LSTM network for arrhythmia classification and clinical diagnostic support. *Biomedical Signal Processing and Control*. 2026;112:108697.
<https://doi.org/10.1016/j.bspc.2025.108697>
16. Lilhore UK, Simaiya S, Alhussein M, Dalal S, Aurangzeb K, Hussain A. An attention-driven hybrid deep neural network for enhanced heart disease classification. *Expert Systems*. 2025;42(2):e13791.
<https://doi.org/10.1111/exsy.13791>
17. Baig Z, Nasir S, Khan RA, Ul Haque MZ. ArrhythmiaVision: Resource-conscious deep learning models with visual explanations for ecg arrhythmia classification. *The Journal of Supercomputing*. 2026;82(8):474.
<https://doi.org/10.1007/s11227-026-08629-9>
18. Farag MM. A tiny matched filter-based CNN for inter-patient ECG classification and arrhythmia detection at the edge. *Sensors*. 2023;23(3):1365.
<https://doi.org/10.3390/s23031365>
19. Seitanidis P, Gialelis J, Papaconstantinou G, Moschovas A. Identification of heart arrhythmias by utilizing a deep learning approach of the ECG signals on edge devices. *Computers*. 2022;11(12):176.
<https://doi.org/10.3390/computers11120176>
20. Al Fahoum A, Ababneh M, Al Qwaider M, Al Zaben A. Real-time embedded detection of bradycardia, tachycardia and arrhythmia using photoplethysmography and on-device convolutional neural networks. *Biomedical Signal Processing and Control*. 2026;121:110345,
<https://doi.org/10.1016/j.bspc.2026.110345>
21. Nugroho H, Peng KJ, Shanmugam A. AI-enhanced ECG analysis for arrhythmia classification with edge device compatibility. *Advances in Computational Intelligence*. 2026;6(1):2.
<https://doi.org/10.1007/s43674-026-00087-3>
22. Liu H. A hybrid model combining 1D-CNN and BERT for intelligent ECG arrhythmia classification. *Scientific Reports*. 2025.
<https://doi.org/10.1038/s41598-025-28023-4>
23. Oporto E, Mauricio D, Maculan N, Uribe G. Challenges in the Classification of Cardiac Arrhythmias and Ischemia Using End-to-End Deep Learning and the Electrocardiogram: A Systematic Review. *Diagnostics*. 2026;16(1):161
<https://doi.org/10.3390/diagnostics16010161>
24. Zubair M, Woo S, Lim S, Kim D. Deep representation learning with sample generation and augmented attention module for imbalanced ECG classification. *IEEE Journal of Biomedical and Health Informatics*. 2023;28(5):2461-72.
doi: 10.1109/JBHI.2023.3325540
25. Shah HA, Saeed F, Diyan M, Almujaally NA, Kang JM. ECG-TransCovNet: A hybrid transformer model for accurate arrhythmia detection using Electrocardiogram signals. *CAAI transactions on intelligence technology*. 2024.
<https://doi.org/10.1049/cit2.12293>

How to Cite: Lokesha HR, Poorani S, Shete P, Vimalapriya MD, Lade SG, Dongre SP, Jegajothi B. Lightweight CNN Architectures for Real-time ECG Arrhythmia Classification on Edge Devices. *Int Res J Multidiscip Scope*. 2026;7(4):649-662. DOI: 10.47857/irjms.2026.v07i04.010430