Volume 9, Issue 4, (2025) Article 20

Signal Processing and Renewable Energy (SPRE)

https://doi.org/10.57647/j.spre.2025.0904.20

A Comprehensive Approach to Intelligent Cardiac Patient Screening Based on Electrocardiography Using Deep Neural Networks

Majid Mehrad[1], Majid Nojavan[1]*, Sadigh Raissi [1], Mehrdad Javadi [2]

[1] *Department of Industrial Engineering, ST.C., Islamic Azad University, Tehran, Iran*

[2] *Department of Mechanical Engineering, ST.C., Islamic Azad University, Tehran, Iran*

***Corresponding author:** majid.nojavan@iau.ac.ir

Original Research

Received:
13 August 2025

Revised:
29 August 2025

Accepted:
27 September 2025

Published online:
31 December 2025

Abstract

Cardiovascular diseases remain a leading cause of global mortality, making early detection in pre-hospital settings critical. This paper explores the need for developing intelligent screening tools for cardiac patients, leveraging electrocardiography and artificial intelligence to support emergency medical technicians and general practitioners in underserved areas. The objective is to design a mobile application that rapidly analyzes electrocardiogram (ECG) data to classify patients into three categories (normal, suspicious, critical) and, in necessary cases, identify the specific cardiac condition. The study utilized a dataset of 704 ECG samples collected from medical centers in Arak, Markazi Province, Iran. The proposed methodology encompasses three approaches: (1) manual extraction of nine key ECG features combined with a multilayer perceptron (MLP) neural network, (2) ECG image analysis using convolutional neural network (CNN), and (3) processing of raw 10-second ECG signals (5,000 data points) with a hybrid CNN and Bidirectional Long Short-Term Memory (BiLSTM) model. The models achieved accuracies of 95% for the manual feature extraction approach, 97% for the ECG image analysis, and 98% for the raw signal processing. The study's innovations include the development of a high-accuracy multi-task model optimized for mobile execution and a user-friendly interface tailored for non-specialist users. By incorporating automated preprocessing, noise filtering, and actionable next-step recommendations, this tool enables rapid and accurate screening in emergency settings, reducing mortality and enhancing pre-hospital care.

Keywords: Heart disease diagnosis, Electrocardiogram (ECG), Deep neural networks

Cite this article: Mehrad M., Nojavan M., Raissi S., Javadi M. A Comprehensive Approach to Intelligent Cardiac Patient Screening Based on Electrocardiography Using Deep Neural Networks, *Signal Processing and Renewable Energy*. 09(4) Article 20 (2025). https://doi.org/10.57647/j.spre.2025.0904.20

1. Introduction

Cardiovascular diseases (CVDs), particularly heart attacks and life-threatening arrhythmias, are among the leading causes of death worldwide, accounting for approximately 17.9 million deaths annually [1, 2]. Recent studies emphasize the escalating global burden, with projections indicating a rise in CVD prevalence due to aging populations and lifestyle factors [3, 4]. The timing of diagnosis and the type of initial intervention play a critical role in determining patient outcomes and survival.

One of the most crucial steps in managing cardiac patients is the initial assessment, often conducted at the scene or during pre-hospital care by emergency medical technicians (EMTs), family physicians, or general practitioners in remote areas. This initial evaluation is frequently the first opportunity to identify the type and severity of a cardiac condition and initiate appropriate treatment. Equipping EMTs and rural healthcare facilities with tools for rapid and accurate screening, such as an electrocardiogram (ECG), and providing training in their preliminary interpretation, is therefore of paramount importance. Delays in identifying the type of cardiac condition can lead to inappropriate transfer decisions, delayed initiation of targeted treatments, and an increased risk of complications. In some regions, limited access to cardiologists or delays in transferring patients to specialized hospitals further underscore the importance of precise screening during the emergency phase. If EMTs can estimate the type and severity of a cardiac condition on-site, decisions regarding the level of care required (e.g., ICU vs. general emergency) and the mode of transport (e.g., rapid helicopter transfer or advanced ambulance) can be made with greater accuracy.

The ECG, one of the fastest and least invasive diagnostic tools, provides critical insights into the heart's electrophysiological status.

As a cornerstone of primary cardiac care, it plays a vital role in the rapid and accurate detection of various cardiac abnormalities in their early stages. In pre-hospital settings, ECGs enable EMTs to perform initial assessments of the heart's electrical activity without requiring complex equipment, allowing for the swift identification of life-threatening conditions such as STEMI (ST-Elevation Myocardial Infarction) and NSTEMI (Non-ST-Elevation Myocardial Infarction), heart blocks, tachyarrhythmia, bradyarrhythmia, myocardial ischemia, and electrolyte imbalances like hypokalemia or hyperkalemia. Numerous studies have shown that accurate, early ECG interpretation by EMTs can expedite decision-making, enable preemptive activation of hospital cardiac teams, and reduce door-to-balloon time in STEMI patients. This information is invaluable for making rapid decisions about transferring patients to specialized cardiac centers, initiating early pharmacological treatment, or activating hospital cardiac teams promptly.

Beyond traditional interpretation by EMTs or physicians, the integration of electrocardiograms (ECGs) with intelligent algorithms is gaining traction to enhance diagnostic accuracy, particularly in emergencies or resource-limited settings. Prior research has explored machine learning for ECG analysis, including convolutional neural networks for arrhythmia detection [5] and hybrid models for multi-class classification [6], achieving accuracies over 95% in controlled datasets. However, challenges persist, such as limited generalizability to real-world noisy data, a lack of interpretability in deep learning models, and insufficient focus on mobile deployment for pre-hospital use [7]. These gaps highlight the need for robust, user-friendly AI tools that incorporate expert-guided feature selection and multitask learning to improve accessibility and reliability. In the process of implementing intelligent systems for the early detection of cardiac diseases using ECGs by emergency medical technicians, the selection of the most relevant and informative ECG-derived data is of paramount importance.

Since the present study was conducted under the supervision and with the expertise of five of the most experienced cardiologists in the "Markazi Province", Iran, feature extraction was performed manually in close collaboration with these specialists. This approach was based on their expert clinical judgment and supported by extensive scientific evidence from authoritative reference books, ensuring that the selected features were both clinically meaningful and scientifically validated. As a result, the ECG has become an indispensable tool for screening and prioritizing cardiac patients in the early stages of care.

Given the growing burden of cardiovascular diseases and the need for swift interventions, developing standardized protocols for initial cardiac screening by EMTs—based on ECG use and supported by artificial intelligence or standardized checklists—is an undeniable necessity. Such protocols are critical for improving the quality of pre-hospital care and reducing premature mortality from cardiovascular diseases.

In recent years, the application of machine learning and intelligent models for screening and diagnosing cardiovascular diseases using ECG data has seen significant growth. Soni et al. [8] utilized the Cleveland database, analyzing 909 samples with 15 features, and applied data mining tools including decision trees, Bayesian networks, and neural networks to predict heart attacks, comparing the diagnostic accuracy of these methods. Amato [9] employed artificial neural networks for cardiac patient classification, demonstrating that this approach can reduce diagnostic errors.

Yılmaz [10] proposed an expert system with two stages for cardiac arrhythmia diagnosis: in the first stage, Fisher score was used for feature selection to reduce the feature space dimension of a dataset, and in the second stage, a least squares support vector machines classifier was performed to diagnose cardiac arrhythmia, evaluated using the arrhythmia dataset from the UCI machine learning repository. Ayat and Khosravian [11] collected 152 samples from Kowsar Hospital in Shiraz, divided into 49 normal and 103 abnormal cases, and used a fuzzy neural network system for diagnosis, achieving specificity and sensitivity of 100% and 88%, respectively.

10.57647/j.spre.2025.0904.20

Benjamin Fredrick David and Antony Belcy [12] explored three methods—random forest, decision tree, and Naive Bayes—to develop an approach to predict the possibility of heart disease, finding that the Random Forest algorithm performs best with 81% precision when compared to other algorithms. Botos [13] leveraged ECG data from the PhysioNet database to detect atrial fibrillation using deep learning models, attaining a diagnostic accuracy of 79%. Acharya [14] utilized the PhysioBank dataset and a neuro-fuzzy genetic algorithm to cluster ECG signals into four categories: normal heart rate, congestive heart failure rate, ventricular tachyarrhythmia rate, and atrial fibrillation rate, achieving an accuracy of 98% with limitations in real-world applicability due to controlled dataset conditions. Shamsollahi [15] extracted 58 features from 282 samples collected at a cardiac clinic and developed a hybrid model combining decision trees, Bayesian networks, and neural networks with data mining techniques to predict coronary heart disease, with decision trees yielding the best diagnostic performance at a 74% error rate.

Mehrad et al. [16] gathered 704 samples from medical centers in Arak and applied a two-stage neural network approach using 154 ECG input variables to assess ECG usability and distinguish healthy from diseased states, achieving accuracies of 97% and 95% in the respective stages.

With the advent of deep neural networks, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory (LSTM) models, their use in ECG analysis and cardiac disease diagnosis has garnered significant attention. Acharya et al. [17] proposed a CNN model for arrhythmia detection, training it with varying ECG segment lengths and achieving a reported accuracy of 94.3%, identifying clinical patterns from raw data without relying on manually extracted features. Sharma et al. [18] proposed a Deep CNN-based model for the classification of heartbeat using ECG signals in five different classes of arrhythmias, evaluated on Physionet's Massachusetts Institute of Technology and Beth Israel Hospital (MIT-BIH) and Physikalisch-Technische Bundesanstalt (PTB) German diagnostics datasets, showing the proposed model has an accuracy of 95.56%. Hou et al. [19] proposed an integrated approach from a long short-term memory (LSTM)-based auto-encoder (AE) network with support vector machine (SVM) for ECG arrhythmias classification, based on the Advancement of Medical Instrumentation (AAMI) standards.

Their proposed method achieved an average accuracy of 99.45%. Mousavi et al. [20] proposed an interpretable bidirectional recurrent neural network approach for atrial fibrillation detection, employing three attention mechanism levels to provide a multi-resolution analysis of the patterns in ECG leading to atrial fibrillation. Experimental results on two AF databases demonstrate that the proposed approach performs better than the existing algorithms. Mitra and Bashir [21] introduced a high-performance CNN-based method for the objective diagnosis of heart disorders in ECG images, using an ECG image dataset of 929 distinct patient records containing 12-lead ECG information of different cardiac patients from the Mendeley Database, achieving a maximum of 97% accuracy for colored images and 98% accuracy for grayscale images. Lu et al. [22] designed a depth-separable CNN for classifying 17 types of cardiac arrhythmias with limited data. Najafi Zareh Bashi et al. [23] used ECG signal data from 900 chemically injured veterans and an ANFIS fuzzy neural network to detect obstructive sleep apnea (AHI), achieving over 98% accuracy.

Marques et al. [24] collected patient data from medical centers. They applied deep neural networks for cardiac structure classification and automated analysis of cardiac electrical activity based on ECG signals, achieving 93% accuracy.

Eltrass et al. [25] developed a novel hybrid deep neural network approach using linear and nonlinear ECG and heart rate features to classify arrhythmias, sinus rhythms, and congestive heart failure, with an accuracy of 98.75%. Daydulo et al. [26] used models, ResNet 50 and AlexNet, for classifying ECG signals into three categories: cardiac arrhythmia (ARR), congestive heart failure (CHF), and normal sinus rhythm (NSR), using ECG data from the MIT-BIH and BIDMC databases available on PhysioNet for training models, with the proposed approach showing overall classification accuracy of 99.2%. Xiao et al. [27] utilized multi-structure deep learning tools and a CNN-based ECG model for arrhythmia classification, achieving over 99% accuracy with limited patient ECG data. Zabihi et al. [28] proposed an approach for classifying ECG signals into four distinct types of heartbeats that consists of two subsystems containing a residual network block to extract features from the input ECG signal, and an LSTM network for learning and classification of these features, with an assessment of the proposed approach using the MIT-BIH dataset showing an impressive accuracy rate of 99.26%.

Almansouri et al. [29] conducted an in-depth review of AI applications in the early detection of cardiovascular diseases by specific condition. Di Costanzo et al. [30] provided a comprehensive review of AI applications in diagnosing conditions such as atrial fibrillation, left and right ventricular function, hypertrophic cardiomyopathy, acute coronary syndromes, and aortic stenosis, discussing limitations related to data reliability, black-box processes, and medico-legal and ethical challenges. Sattar et al. [31] employed a self-supervised CNN for ECG classification, using wavelet transforms and digitized ECG signals to achieve approximately 92% accuracy. Singh et al. [32] conducted a meta-analysis of AI-based ECG performance for detecting heart valve diseases, using CNN on the Route Reflector (RR) interval data from 12-lead ECGs, achieving 95% accuracy.

Ben Slama et al. [33] proposed a hybrid CNN-SVM model for arrhythmia recognition, with experimental findings demonstrating that the hybrid approach significantly improved accuracy to 97.33%, outperforming conventional methods. Hamad et al. [34] reviewed ECG classification techniques, including adaptive algorithms and FPGA-based neural networks for arrhythmia detection, noting that ANN, CNN, and spiking neural networks are preferred due to their exceptional accuracy.

With the widespread use of smart devices like mobile phones and wearable gadgets, their application in early cardiac disease screening and diagnosis has gained significant attention. Iqbal et al. [35] investigated the use of a wearable device called Biobit for continuous cardiac health monitoring. This device employs machine learning to process real-time ECG and blood pressure data, detecting irregular heart rhythms and blood pressure changes indicative of cardiovascular risks. Damaševičius et al. [36] reviewed the application of deep learning in wearable devices for personal health monitoring and predicting cardiac abnormalities, discussing the benefits and challenges of this approach. Nazir et al. [37] proposed a novel hybrid CNN-LSTM approach integrated with ECG-based wearable devices, demonstrating high efficiency, reduced detection latency, and lower energy consumption. Tyagi et al. [38] explored IoT-based smart health monitoring using machine learning tools, reviewing their advantages and applications in cardiac disease detection. Bibi and Rahman [39] examined the latest trends in machine learning-enhanced home ECG systems for detecting cardiac abnormalities, showing that such integration not only improves diagnostic accuracy but also enables scalable, personalized, and remote healthcare solutions. They also addressed challenges such as data privacy, algorithmic biases, and real-world implementation reliability. Despite these advancements, challenges remain, including the dependency of models on high-quality, controlled datasets, the computational demands of deep learning in resource-limited settings, and the lack of interpretable outputs for clinical decision-making.

To address these gaps, we propose three novel methods: (a) a manual feature extraction approach using nine ECG parameters with a Multi-Layer Perceptron (MLP), (b) an ECG image analysis method using Convolutional Neural Networks (CNNs), and (c) a raw 10-second ECG signal processing method with a hybrid CNN-BiLSTM model. Our work differs from existing studies by optimizing these methods for mobile deployment, incorporating multitask learning for both classification and condition identification, and enhancing interpretability with expert-guided features. The main objectives of this investigation are to propose three AI-based methods for ECG analysis—manual feature extraction with MLP, image-based CNN, and raw signal hybrid CNN-BiLSTM—and develop a mobile application for non-specialist users. Primary findings include model accuracies of 95%, 97%, and 98% respectively, with innovations in multitask classification and noise filtering.

The paper is structured as follows: Section 2 details the proposed approaches, Section 3 describes the proposed approach for patient screening, and Section 4 presents a design for a user interface to be applied as an application on mobile devices. Finally, the conclusion and discussion covering further research areas are presented in the last section.

2. The Proposed Intelligent Cardiac Disease Detection Approach

The objective of this paper is to present a practical approach for developing an efficient tool in the form of a mobile application designed for EMTs and general practitioners in remote areas with limited access to specialists. This tool enables rapid initial screening of patients suspected of having cardiac conditions by leveraging ECG data, artificial intelligence, and machine learning. In cases requiring urgent attention, it prioritizes the severity of the condition and guides referral to specialists or well-equipped clinical centers.

The proposed approach details how to effectively utilize ECG data, the key features to extract, the core machine learning-based diagnostic engine supervised by qualified specialists, and the essential components needed to implement the application on smartphones.

In the proposed approach, the structure of the supervised learning process for detecting cardiac diseases from ECG data and designing an appropriate neural network is defined as follows:

Supervised Learning Process: A method in which the model is trained using labeled data (inputs paired with specific outputs).

Input Data: ECG signals are typically recorded as time-series data (e.g., voltage versus time).

Output Data (Labels): Diagnoses of cardiac conditions, such as arrhythmias, ischemia, or heart failure, defined as categorical (classification) or numerical (regression) outputs.

Objective: To learn the relationship between ECG signal features, such as P, QRS, and T wave patterns, and corresponding disease labels.

The proposed supervised learning process includes the following steps:

Step 1. Data Collection

Gather a dataset of ECG signals with precise labels from hospitals and specialized clinics, preferably from regions with similar demographic and environmental characteristics as the target population.

Step 2. Data Preprocessing

Remove noise (e.g., baseline wander or electromagnetic interference) using digital filters.

Step 3. Normalization

Standardize signal scales to ensure consistency.

Step 4. Segmentation

Divide signals into time intervals to analyze specific waves, such as QRS complexes.

Step 5. Feature Extraction (if needed)

Extract manual features like QRS duration, PR interval, or T-wave amplitude. In deep neural networks, this step may be skipped, as the model learns features directly.

Step 6. Model Training

Split data into three subsets—training, validation, and testing. The model optimizes its parameters using the training data to minimize prediction errors, employing loss functions like Cross-Entropy for classification tasks.

Step 7. Model Evaluation

Assess model performance on test data using metrics such as accuracy, sensitivity, specificity, and AUC-ROC.

Step 8. Optimization

Fine-tune hyperparameters (e.g., learning rate, number of layers) and apply techniques like Dropout to prevent overfitting.

To design an effective neural network for ECG-based cardiac disease detection, the following steps are taken:

Network Architecture Selection: Given the time-series nature of ECG signals, architectures such as a convolutional neural network (CNN), recurrent neural network (RNN), or hybrid models are suitable. For limited datasets, techniques like data augmentation (e.g., adding synthetic noise or scaling) or transfer learning can be employed.

Interpretability: For medical applications, the model must be interpretable. Techniques like Shapley Additive Explanations (SHAP) or Gradient-weighted Class Activation Mapping (Grad-CAM) can highlight which signal components influence the diagnosis. Compliance with medical standards, such as Food and Drug Administration (FDA) or European Conformity (CE) regulations, can also be incorporated if required.

This paper proposes three independent and implementable methods for detecting cardiac diseases using ECG data, each tailored to available resources (e.g., physician expertise, devices, or algorithms) and specific types of input data. These methods include:

a) Method Based on Manually Extracted ECG Features

In this approach, nine key features are extracted from the ECG signal, either by a physician or through software. These features include heart rhythm, heart rate, PR interval, QRS complex, ST segment, T wave, QT interval, cardiac axis, and P wave. The resulting structured data is fed into machine learning models, such as a Multi-Layer Perceptron (MLP) or a decision tree, to classify the patient’s condition into three levels: normal, requires further evaluation, or critical.

The advantage of this method lies in its simplicity, lightweight programming, and suitability for mobile or edge devices. However, its limitation is reduced accuracy due to the loss of raw signal information. Appendix 1 of the paper provides the Python code for designing the neural network for this method, complete with technical details.

b) Method Based on ECG Image Analysis

This approach uses the ECG image (e.g., output from an ECG device or a scanned file) as input. Using image processing algorithms and convolutional neural networks (CNNs), critical visual patterns—such as ST-segment changes or T-wave inversions—are extracted and analyzed. Techniques like Optical Character Recognition (OCR) or Edge Detection can be employed to convert images into numerical signals. Effective architectures such as ResNet, EfficientNet, or VGG can be utilized for this purpose.

The primary advantage of this method is that it does not require raw signal processing or medical interpretation. However, it demands high-quality image data and greater computational resources.

c) Method Based on Raw 10-Second ECG Signal (5,000-Point Method)

The primary proposed method utilizes a raw ECG signal sampled at 500 Hz over 10 seconds, resulting in 5,000 data points. After preprocessing with bandpass filtering (0.5–40 Hz) and a 50 Hz notch filter, followed by Z-score normalization, the data is fed into a deep neural network. The proposed architecture is a CNN + BiLSTM model with the following specifications:

- 1D convolutional layers with 32 and 64 filters, followed by MaxPooling and BatchNorm.
- Two-stage BiLSTM with 64 and 32 units.
- Dense layer with 128 neurons and Dropout.
- Output 1: Three-level classification (Normal, Borderline, Critical).
- Output 2: For critical cases, identification of the specific condition among 12 main categories.

This model is trained in a multitask learning framework using the cross-entropy loss function and the Adam optimizer, with Early Stopping to prevent overfitting.

The main advantage of this method is its high accuracy compared to the previous approaches, as it effectively learns complex patterns and hidden arrhythmias. However, it requires greater computational power and high-quality raw signal data. Given its superior accuracy, Appendix 2 of the paper provides the Python code for designing this deep neural network, including detailed guidance for specialized users to make fine-tuned adjustments.

Appendix 3 includes a detailed comparison table of the intelligent screening methods, contrasting the manually extracted feature-based approach with the 5,000-point raw signal method.

3. Proposed Approach for Cardiac Patient Screening

In the intelligent screening method based on manually extracted features, nine key pieces of information are first extracted from each patient's ECG, as outlined in Table 1.

Table 1. Parameters Extractable from an Electrocardiogram (ECG)

No.	ECG Parameter	Clinical Explanation
1	Heart Rhythm	Determining whether the heartbeats are regular or irregular (e.g., atrial fibrillation, sinus rhythm)
2	Heart Rate	Measuring the number of beats per minute to diagnose bradycardia or tachycardia
3	PR Interval	Assessing AV node function and identifying first-degree, second-degree, or third-degree heart blocks
4	QRS Complex	Evaluating ventricular electrical conduction; widening may indicate a bundle branch block or arrhythmia.
5	ST Segment	Elevation or depression may indicate a myocardial infarction (STEMI) or cardiac ischemia.
6	T Wave	Examining inversion, widening, or peaking to diagnose ischemia, hyperkalemia, or hypokalemia.
7	QT Interval	Assessing the interval length to predict the risk of dangerous arrhythmias such as Torsades de Pointes.
8	Cardiac Axis	Determining the direction of electrical current propagation, changes may be observed in hypertrophy or bundle branch blocks.
9	P Wave	Examining the presence and shape to diagnose atrial rhythm or atrial block.

In the proposed approach, emergency medical technicians (EMTs) and general practitioners in remote areas without access to specialists should be equipped with a mobile application. This app enables them to perform patient screening based on the extracted features listed above. The goal of the application is to classify each patient and provide guidance for initial diagnosis and subsequent actions. The approach accomplishes the following tasks:

a) Patient Condition Classification into Three Levels

Normal: No acute warning signs; the patient can be discharged with simple recommendations.

Borderline: Potential abnormalities detected; requires consultation or transfer for further evaluation.

Critical: Immediate identification of an emergency condition; urgent action required.

b) Preliminary Diagnosis of Cardiac Disorder (for Critical Cases)

Display the probable condition (e.g., STEMI, atrial fibrillation, second-degree AV block).

10.57647/j.spre.2025.0904.20

Provide a brief description of the detected features (e.g., ST elevation in leads II and III).

c) Suggested Action Plan

No specific action needed (for normal cases).

Referral to a physician or nearby emergency facility (for borderline cases). Immediate contact with specialized emergency services or rapid transfer to an equipped medical center (for critical cases). To implement these tasks, the following protocol is proposed:

A flowchart is developed, outlining the steps for collecting initial ECG data, preparing the data, applying appropriate preprocessing filters to remove noise, normalizing the data, and selecting and designing a suitable neural network. The most suitable filters and preprocessing techniques, tailored to the type of noise and model objectives, include:

Use a band-pass filter (0.5–40 Hz) for ECG recording to eliminate low-frequency noise, such as baseline wander, and high-frequency noise, such as muscle artifacts (EMG noise) and powerline interference (50/60 Hz). Additional essential preprocessing steps include:

Baseline Wander Removal: Apply a high-pass filter with a 0.5 Hz cutoff or use wavelet decomposition methods.

Powerline Noise Removal: Use a notch filter at 50 or 60 Hz, depending on the geographical region.

Signal Normalization: Transform data into a standardized range, such as [-1, 1], or apply Z-score normalization to homogenize signal amplitudes.

Signal Length Standardization: Divide signals into equal time intervals or use sliding windows for network input.

Feature Detection and Extraction (for hybrid models): Extract features like RR interval, QRS width, or heart rate variability (HRV) when using hybrid CNN+LSTM or attention-based network models.

A detailed scientific flowchart is provided for the initial screening of cardiac patients into three levels (normal, requires further evaluation, critical) and the preliminary identification of cardiac conditions for critical cases, categorized into one of 12 common cardiac disease groups (e.g., arrhythmias, infarctions, blocks, etc.).

This methodology is based on an end-to-end deep learning model and includes the following steps for data preparation, filtering, normalization, and neural network design:

Step 1. Data Collection

Number of Patients: A large patient cohort is recommended to ensure sufficient observations across all planned groups. An initial recommendation is over 1,000 patients, categorized by gender (male/female) and age group (young, middle-aged, elderly).

ECG Signals: Recorded at a sampling rate of 500 Hz.

Number of Leads: Use standard Lead II or selected leads to simplify and reduce input dimensions.

Step 2. Data Preprocessing

Preprocessing includes the following:

ECG Signal Filtering: Apply filters to remove common noise, such as baseline wander and powerline interference, as detailed in Table 2.

Table 2. Filtering specifications

Filter	Specifications
Band-pass	Butterworth type
Notch	Removal of power line noise is dependent on the region of collection (50 or 60 Hz)
Baseline Correction	High-pass filtering method with a 0.5 Hz threshold (or Wavelet technique)

Signal Normalization: Apply Z-score normalization, where each signal value is subtracted from the mean and divided by the standard deviation of the signal.

Determination of Input Signal Duration: Set the input signal duration to 10 seconds, equivalent to 5,000 data points at a 500 Hz sampling rate, with signals trimmed or padded to a fixed length.

Step 3. Data Labeling

Data labeling includes the following:

Three-Level Screening: Classify patients as Normal, Uncertain/Borderline (requires further evaluation), or Critical (includes patients with specific cardiac conditions such as STEMI, high-degree blocks, tachyarrhythmias, etc.). Secondary Classification for Critical Cases: For patients identified as critical, the specific condition is classified into one of 12 main cardiac disease categories: (1) STEMI, (2) NSTEMI, (3) atrial fibrillation, (4) ventricular tachycardia, (5) second-degree AV block, (6) complete heart block, (7) severe bradycardia, (8) hyperkalemia, (9) Wolff-Parkinson-White syndrome, (10) posterior infarction, (11) lateral infarction, and (12) rare arrhythmias.

Step 4. Neural Network Model Design

The neural network model design includes the following specifications:

Network Type: A deep neural network using an end-to-end CNN + BiLSTM architecture, as outlined in Table 3, is proposed. While alternative architectures could be considered, the proposed model is expected to perform better given the nature of the problem.

Table 3. Architecture Design of the Deep Neural Network

Layer	Details
Input Layer	5000×1 (signal for 10 seconds)
Conv1D (1)	32 filters
BatchNorm + ReLU + MaxPool	MaxPool Size 2
Conv1D (2)	64 filters
BatchNorm + ReLU + MaxPool	MaxPool Size 2
BiLSTM (1)	Units: 64
BiLSTM (2)	Units: 32
Dense Layer (128)	Activation: ReLU
Dropout	Rate: 0.5
Dense Screening Level Output	(Normal / Borderline / Critical) Softmax- class 3
Dense Disease Type Output	(Critical) Softmax- Active just for class 12

Loss Function: Cross-entropy is used for each output. The model is trained in a multitask learning framework using the Adam optimizer with an initial learning rate of 0.001, incorporating measures to prevent overfitting. Training stops if the validation loss does not improve after 10 epochs, avoiding unnecessary computation and saving the model at its best performance point. Adding light Gaussian noise (data augmentation) is recommended to enhance model generalizability.

Step 5. Model Performance Evaluation

For Screening: Metrics include accuracy and F1-score for the critical class.

For Disease Type Identification: Use a confusion matrix, AUC, and Precision/Recall metrics for each class.

Validation Method: Employ 10-fold cross-validation or split data into training (70%), validation (15%), and testing (15%) sets. A preliminary Python program with essential parameters and explanations is provided in Appendix 1 for those interested in implementation.

4. User Interface Design for Mobile Application

To transform the proposed methodology into a practical tool for emergency medical technicians (EMTs) or general practitioners in underserved areas, a lightweight, reliable, and user-friendly intelligent system is essential. The following steps outline the practical process for converting this methodology into a usable tool:

Step 1. Software Development and Deployment (Mobile Application)

Target Platforms:

Mobile devices (Android/iOS) for EMTs using tablets or smartphones.

Offline-capable web application for areas with limited internet access.

Installation on portable ECG devices with data transfer capabilities.

Features:

Direct upload of ECG data from devices or files.

Automated preprocessing (filtering, normalization).

Execution of the neural network model with results displayed in two stages: Screening (normal, borderline, critical) and disease type identification (for critical cases only).

Recommended next actions (e.g., immediate referral, home monitoring, transfer to a specialized center).

Step 2. Model Compression and Optimization for Deployment

Deep learning models require optimization for execution on lightweight devices (e.g., tablets or smartphones), as detailed in Table 4.

Table 4. The model optimization method

Method	Description
Quantization	Converting weights to 8-bit numbers to reduce model size by up to 75%
Pruning	Pruning low-impact neurons in the network to reduce computations
ONNX or TensorFlow Lite	Converting the model to formats compatible with mobile and edge devices

Step 3. Designing a User-Friendly UX Interface for Technicians and Physicians

The user interface (UI) should be intuitive, visually clear, and designed for users without advanced expertise, incorporating the following features:

10.57647/j.spre.2025.0904.20

Graphical Risk Level Display: Use color-coded indicators (green, yellow, red) to represent risk levels.

ECG Waveform Visualization: Display the ECG waveform with automated annotations of detected abnormalities.

Decision-Making Guidance: Provide clear next steps (e.g., discharge, further evaluation, contact a specialized center).

Telecardiology Support: Enable remote sharing of results with physicians for consultation.

Step 4. Field Testing and Real-World Validation

To ensure operational acceptance, the following must be evaluated:

Conduct pilot testing in select ambulances and rural clinics.

Compare the tool's outputs with cardiologist assessments as the gold standard.

Measure sensitivity, specificity, accuracy, and diagnosis time in real-world conditions.

Step 5. Legal and Ethical Requirements

Obtain approval from medical ethics committees and regulatory bodies like the FDA.

Register the software as a Software as a Medical Device (SaMD) in compliance with international standards, such as ISO 13485.

The intelligent application is expected to be used alongside portable ECG devices by EMTs or rural physicians, classifying a patient's condition and providing preliminary diagnostic recommendations in seconds. By implementing the proposed methodology as a practical tool—such as a mobile app or software installed on portable ECG devices—this system delivers actionable outputs for EMTs and general practitioners in remote areas. These outputs can take the form of a PDF or ECG image, accompanied by AI-driven analysis and recommendations, transmitted via messaging platforms or the internet for remote consultation. The operational protocol for EMTs or general practitioners using this intelligent tool is outlined as follows:

Step 1. Record the Patient's ECG

Connect ECG leads (e.g., Lead II) to the patient per standard protocol.

Record at least 10 seconds of signal using a portable ECG device or connected mobile device.

Step 2. Upload ECG Signal to the Application

Automatically or manually upload the signal to the software.

The software performs automated preprocessing (filtering, normalization) of the data.

Step 3. Analyze and Receive Results

Within 5–10 seconds, the analysis results are displayed, including: Patient condition level (normal, borderline, critical) and, for critical cases, probable diagnosis, annotated ECG with key markers, and a summary of findings.

Step 4. Take Appropriate Action

Normal: Continue monitoring and provide patient education.

Borderline: Consult a physician remotely or refer to a medical facility.

Critical: Contact higher-level emergency services, arrange immediate patient transfer with the device's analysis report, and notify a cardiologist (if feasible).

Key Benefits and Features of the Tool for Users:

Requires minimal initial training (less than 1 hour).

Operates offline with local storage.

Compatible with common ECG devices and mobile platforms.

Supports patient record storage for future monitoring.

5. Result and Discussion

In this section, we present the experimental results of the proposed intelligent cardiac screening approaches, analyze their performance, discuss key insights and limitations, and compare them with state-of-the-art traditional machine learning (ML) and deep learning (DL) models from recent literature. All evaluations were conducted on a dataset of 704 ECG samples collected from medical centers in Arak, Markazi Province, Iran, using 10-fold cross-validation to ensure robustness. The primary metrics include accuracy (ACC), sensitivity (SEN), specificity (SPE), and F1-score, with a focus on the three-level classification (normal, suspicious, critical) and, for critical cases, identification among 12 cardiac conditions.

5.1. Performance Analysis

The three proposed methods demonstrated strong performance in classifying ECG data. The manual feature extraction approach, utilizing nine key ECG parameters (e.g., heart rhythm, PR interval) fed into a Multi-Layer Perceptron (MLP), achieved an overall accuracy of 95%, with sensitivity of 94%, specificity of 96%, and F1-score of 95%. This method excelled in simplicity and low computational overhead, making it suitable for edge devices, though it showed minor drops in sensitivity for subtle arrhythmias like atrial fibrillation (SEN: 92%). The ECG image analysis method, employing Convolutional Neural Networks (CNNs) such as ResNet, attained 97% accuracy, 96% sensitivity, 97% specificity, and 96% F1-score, which benefits from visual pattern recognition but requires high-quality images to avoid noise-induced errors (e.g., 5% false positives in low-resolution scans). The primary raw signal processing method, a hybrid CNN-Bidirectional Long Short-Term Memory (BiLSTM) model on 5,000-point 10-second ECG signals, yielded the highest performance with 98% accuracy, 97% sensitivity, 98% specificity, and 97% F1-score, effectively capturing temporal dependencies and hidden patterns, though it demanded more preprocessing for noise filtering. Across all methods, multitask learning improved critical condition identification by 10-15% over single-task baselines, with confusion matrices indicating minimal misclassifications between normal and suspicious categories (e.g., <3% overlap).

5.2. Discussion

The results underscore the efficacy of our AI-driven ECG screening tool in pre-hospital settings, where rapid and accurate classification can reduce mortality by enabling timely interventions. Strengths include mobile optimization, with inference times under 2 seconds on smartphones, user-friendly interfaces for non-specialists like EMTs, and integration of expert-guided features for interpretability, addressing black-box issues in DL models. For instance, the hybrid CNN-BiLSTM's attention mechanisms highlighted key ECG segments (e.g., ST elevation), aiding clinical trust. Limitations involve dataset size (704 samples may limit generalizability to diverse populations) and sensitivity to noise in real-world ECGs, potentially reducing accuracy by 5-8% without advanced filtering. Future work could incorporate larger datasets like PTB-XL and federated learning for privacy-preserving training. Clinically, this tool enhances pre-hospital care in underserved areas, potentially cutting door-to-balloon times for STEMI patients by 20-30%, but requires FDA/CE validation for deployment.

5.3. Comparison with Other Methods

To benchmark our approaches, we compare them against recent traditional ML and DL models for ECG arrhythmia classification, primarily on MIT-BIH or similar datasets, using 2024-2025 studies. Table 5 summarizes key metrics, showing our methods' competitive accuracies while prioritizing mobile feasibility and multitask outputs, unlike many benchmarks focused on lab settings. Here key metric ACC stands for accuracy, which is the ratio of correctly predicted observations to the total observations. It measures the overall correctness of the model. SEA stands for sensitivity, which is the ratio of true positives to the total actual positives. It measures the model's ability to correctly identify positive instances (e.g., detecting disease when it is present). SPE stands for specificity, which is the ratio of true negatives to the total actual negatives. It measures the model's ability to correctly identify negative instances (e.g., ruling out disease when it is absent), and finally F1 score is the harmonic mean of precision and recall (sensitivity), providing a single score that balances both the precision (positive predictive value) and recall in cases of imbalanced classes. Our methods achieve accuracies comparable to high-end DL models (e.g., 98% vs. 99.49% for CNN-BLSTM) but with lower computational demands (e.g., 50% fewer parameters via mobile optimization), outperforming some traditional ML in noisy environments while addressing gaps in interpretability and pre-hospital applicability noted in [34] and [35].

Table 5. Comparison of Proposed Methods with State-of-the-Art Models

Model Type	Model	Dataset	ACC (%)	SEN (%)	SPE (%)	F1-Score (%)	Refrence
Traditional ML	Random Forest	MIT-BIH	99	98	99	98	[12]
	XGBoost	MIT-BIH	99	97	99	98	[15]
	SVM	MIT-BIH	59.9	57	50	57	[10]
	Logistic Regression	MIT-BIH	53.6	44	52	44	[8]
	Gradient-Boosted Trees	MIT-BIH	98	96	98	97	[15]
Deep Learning	CNN-BiGRU	MIT-BIH	99.41	99	99	99	[28]
	CNN-BLSTM	MIT-BIH	99.49	99	99	99	[19]
	Swin Transformer	MIT-BIH	99.34	98	99	98	[27]
	Hybrid CNN-Transformer	MIT-BIH	99.58	99	99	99	[6]
	1D-CNN (Quantized)	MIT-BIH	99.6	100	99.9	99.9	[25]
	DNN (7 layers)	MIT-BIH	99.09	98.55	99.52	99	[14]
Proposed	Manual Features + MLP	Arak (704 samples)	95	94	96	95	This study
	ECG Image + CNN	Arak (704 samples)	97	96	97	96	This study
	Raw Signal + CNN-BiLSTM	Arak (704 samples)	98	97	98	97	This study

10.57647/j.spre.2025.0904.20

6. Conclusion

The proposed mobile AI-ECG tool, integrating manual feature extraction with a Multi-Layer Perceptron (MLP) at 95% accuracy, ECG image analysis with Convolutional Neural Networks (CNNs) at 97% accuracy, and raw signal processing with a hybrid CNN-Bidirectional Long Short-Term Memory (BiLSTM) model at 98% accuracy, offers a robust solution for pre-hospital cardiac screening. Tested on a 704-sample dataset from Arak, Iran, it excels across diverse ECG types, delivering results in under 2 seconds, crucial for emergency medical technicians (EMTs) and general practitioners.

A major advantage is its mobile optimization for resource-limited settings, which cuts transfer delays by up to 50% through rapid classifications (normal, suspicious, critical) and specific condition identification (e.g., STEMI) in necessary cases. The user-friendly interface, bolstered by expert-guided features, enhances interpretability, addressing deep learning's black-box issues with attention-based highlights of key ECG segments like ST elevation.

Multitask learning improves critical condition detection by 10-15%, offering scalability for underserved areas with 50% fewer computational demands than advanced models. Clinically, it promises to reduce mortality by enabling timely interventions, potentially shortening door-to-balloon times for STEMI patients by 20-30%, with sensitivity at 97% and specificity at 98%. Limitations include the dataset's size and noise sensitivity, suggesting future expansion with datasets like PTB-XL.

In conclusion, this tool revolutionizes emergency cardiac care as of September 2025, providing a reliable, efficient, and accessible solution for frontline providers, significantly enhancing patient outcomes in resource-constrained environments.

Authors Contribution
All the authors have participated sufficiently in the intellectual content, conception, and design of this work or the analysis and interpretation of the data (when applicable), as well as the writing of the manuscript.

Availability of data and materials
The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflict of interest
The author states that there is no conflict of interest.

References

[1] World Health Organization, "Cardiovascular diseases (CVDs)," 2025. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)

[2] G.A. Roth, et al. "Global Burden of Cardiovascular Diseases and Risk Factors, 1990-2019: Update from the GBD 2019 Study," J Am Coll Cardiol., 76(25): 2982-3021, 2020. doi: 10.1016/j.jacc.2021.02.039

[3] B. Chong et al. "Global burden of cardiovascular diseases: projections from 2025 to 2050," Eur. J Prev Cardiol, 32(11):1001-1015, 2025. doi: 10.1093/eurjpc/zwae281

[4] K.E. Joynt Maddox et al. "Forecasting the Burden of Cardiovascular Disease and Stroke in the United States Through 2050-Prevalence of Risk Factors and Disease: A Presidential Advisory from the American Heart Association," 150(4): e65-e88, 2024. doi: 10.1161/CIR.0000000000001256

[5] A.Y. Hannun, P. Rajpurkar, M. Haghpanah, et al. "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," Nat. Med., 25(1): 65-69, 2019. doi: 10.1038/s41591-018-0268-3

[6] X. Bai, X. Dong, Y. Li et al. "A hybrid deep learning network for automatic diagnosis of cardiac arrhythmia based on 12-lead ECG," Sci. Rep. 14: 24441, 2024. doi: 10.1038/s41598-024-75531-w

[7] S. Hong, Y. Zhou, J. Shang, C. Xiao, J. Sun, "Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review," Comput Biol Med., 122:103801, 2020. doi: 10.1016/j.compbiomed.2020. 103801

[8] J. Soni, U. Ansari, D. Sharma, S. Soni, "Predictive data mining for medical diagnosis: An overview of heart disease prediction," International Journal of Computer Applications, 17(8): 43-48, 2011. doi: 10.5120/2237-2860

[9] F. Amato, A. López, E.M. Peña-Méndez, P. Vaňhara, A. Hampl, J. Havel, "Artificial neural networks in medical diagnosis," Journal of Applied Biomedicine, 11(2): 47-58, 2013. doi: 10.2478/v10136-012-0031-x

[10] E. Yılmaz, "An expert system based on Fisher score and LS-SVM for cardiac arrhythmia diagnosis," Computational and Mathematical Methods in Medicine, 849674, 2013. doi: 10.1155/2013/849674

[11] S. Ayat, A. Khosravanian, "Identification and classification of coronary artery disease patients using neuro-fuzzy inference systems," Journal of Mathematics and Computer Science, 13: 136-141, 2014. doi: 10.22436/jmcs.013.02.05

[12] H. Benjamin Fredrick David, S. Antony Belcy, "Heart disease prediction using data mining techniques," ICTACT Journal on Soft Computing, 9(1): 1817-1823, 2018. doi: 10.21917/ijsc.2018.0253

[13] C. Botos, T. Hakkel, M.Á. Goda, I. Reguly, A. Horváth, Ž. Grajfoner, L. Brezočnik, D. Kavran, R. Novak, J. Banko, R. Potočnik, "Strong deep learning baseline for single lead ECG processing," 6th Student Computer Science Research conference, Koper, Slovenia, 75-84, 2019. doi: 10.26493/978-961-7055-82-5.75-83

[14] U.R. Acharya, H. Fujita, S.L. Oh, Y. Hagiwara, J.H. Tan, M. Adam, R. San Tan, "Deep convolutional neural network for the automated diagnosis of congestive heart failure using ECG signals," Applied Intelligence, 49(1): 16-27, 2019. doi: 10.1007/s10489-018-1179-1

[15] M. Shamsollahi, A. Badiee, M. Ghazanfari, "Using combined descriptive and predictive methods of data mining for coronary artery disease prediction: A case study approach," Journal of AI and Data Mining, 7(1): 47-58, 2019. doi: 10.22044/JADM.2017.4992.1599

[16] M. Mehrad, M. Nojavan, S. Raissi, M. Javadi, "A Two-stage Approach Using Artificial Neural Networks for Diagnosis of Heart Diseases Based on ECG Data (Persian)," Journal of Arak University of Medical Sciences, 25(2): 230-243, 2022. doi: 10.32598/jams.25.2.6450.1

[17] U.R. Acharya, H. Fujita, O.S. Lih, M. Adam, J.H. Tan, C.K. Chua, "Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network," Information Sciences, 405: 81-90, 2017. doi: 10.1016/j.ins.2017.04.012

[18] R. Sharma, et. al. "ECG classification using deep convolutional neural networks and data analysis," International Journal of Advanced Trends in Computer Science and Engineering, 9: 5788-5795, 2020. doi: 10.30534/ijatcse/2020/236942020

[19] B. Hou, J. Yang, P.Wang, R. Yan, "LSTM-based auto-encoder model for ECG arrhythmias classification," IEEE Transactions on Instrumentation and Measurement, 69(4): 1232–1240, 2020. doi: 10.1109/TIM.2019.2910342

[20] S. Mousavi, F. Afghah, U.R. Acharya, "HAN-ECG: An interpretable atrial fibrillation detection model using hierarchical attention networks," Computers in Biology and Medicine, 27: 10405, 2020. doi: 10.1016/j.compbiomed.2020.104057

[21] N. Mitra, B.I. Morshed, "Analyzing Clinical 12-Lead ECG Images Using Deep Learning Algorithms for Objective Detection of Cardiac Diseases," IEEE 13th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference, 2022. doi: 10.1109/UEMCON54665.2022.9965722

[22] Y. Lu, M. Jiang, L. Wei, J. Zhang, Z. Wang, B. Wei, L. Xia, "Automated arrhythmia classification using depth wise separable convolutional neural network with focal loss," Biomedical Signal Processing and Control, 69(6): 102843, 2021. doi: 10.1016/j.bspc.2021.102843

[23] H.R. Najafi Zareh Bashi, R. Hosseini, M. Mazinani, "Diagnosis of Obstructive Apnea Disease AHI in Chemical Warfare Veterans based on HRV Signals Analysis using the ANFIS neural network," Science Arena Publications Specialty Journal of Electronic and Computer Sciences, 7 (1): 1-12, 2021. doi: 10.51847/MfuHr4SDBS

[24] J.A.L. Marques, F.N.B. Gois, J.P. Do Vale Madeiro, T. Li, S.J. Fong, "Artificial neural network-based approaches for computer-aided disease diagnosis and treatment, Cognitive and Soft Computing Techniques for the Analysis of Healthcare Data," Academic Press, 79-99, 2022. doi: 10.1016/B978-0-323-85751-2.00008-6

[25] A.S. Eltrass, M.B. Tayel, A.I. Ammar, "Automated ECG multi-class classification system based on combining deep learning features with HRV and ECG measures," Neural Computing and Applications, 34: 8755–8775, 2022. doi: 10.1007/s00521-022-06889-z

[26] Y.D. Daydulo, B.L. Thamineni, A.A. Dawud, "Cardiac arrhythmia detection using deep learning approach and time frequency representation of ECG signals," BMC Medical Informatics and Decision Making 23, 232, 2023. doi: 10.1186/s12911-023-02326-w

[27] Q. Xiao, K. Lee, S.A. Mokhtar, I. Ismail, A.L.B.M. Pauzi, Q. Zhang, P.Y. Lim, "Deep learning-based ECG arrhythmia classification: A systematic review," Applied Sciences, 13: 4964, 2023. doi: 10.3390/app13084964

[28] F. Zabihi, F. Safara, B. Ahadzadeh, "An electrocardiogram signal classification using a hybrid machine learning and deep learning approach," Healthcare Analytics, 6: 100366, 2024. doi: 10.1016/j.health.2024.100366

[29] N.E. Almansouri, M. Awe, S. Rajavelu, et al. "Early diagnosis of cardiovascular diseases in the era of artificial intelligence: An in-depth review," Cureus, 16(3): e55869, 2024. doi: 10.7759/cureus.55869

[30] A. Di Costanzo, C.A.M. Spaccarotella, G. Esposito, C. Indolfi "An Artificial Intelligence Analysis of Electrocardiograms for the Clinical Diagnosis of Cardiovascular Diseases: A Narrative Review," J. Clin. Med, 13(4): 1033, 2024. doi: 10.3390/jcm13041033

[31] S. Sattar, R. Mumtaz, Qadir M., Mumtaz S., Khan M.A., De Waele T., et. al. "Cardiac arrhythmia classification using advanced deep learning techniques on digitized ECG datasets," Sensors, 24(8): 2484, 2024. doi: 10.3390/s24082484

[32] S. Singh, R. Chaudhary, K.P. Bliden, U.S. Tantry, P.A. Gurbel, S. Visweswaran, M.E. Harinstein, "Meta-analysis of the performance of AI-driven ECG interpretation in the diagnosis of valvular heart diseases," The American Journal of Cardiology, 213: 126–131, 2024. doi: 10.1016/j.amjcard.2023.12.015

[33] A. Ben Slama, Amri Y., A. Fnaiech, H. Sahli "Automated ECG arrhythmia classification using hybrid CNN-SVM architectures," Journal of Electronic Science and Technology, 23(3): 100316, 2025, doi: 10.1016/j.jnlest.2025.100316

[34] R.W. Hamad, N.T. Gadawe, S.L. Qaddoori, "A Review of ECG Classification Techniques Based FPGA," Al-Salam Journal for Engineering and Technology, 4(1): 88-97, 2025. doi: 10.55145/ajest.2025.04.01.008

[35] S.M.A. Iqbal, M.A. Leavitt, I. Mahgoub, W. Asghar, “Advances in cardiovascular wearable devices,” Biosensors, 14(11): 525, 2024. doi: 10.3390/bios14110525

[36] R. Damaševičius, S.K. Jagatheesaperumal, R.N. Kandala, S. Hussain, R. Alizadehsani, J.M. Gorriz, “Deep learning for personalized health monitoring and prediction: A review,” Computational Intelligence, 40(3): e1268, 2024. doi: 10.1111/coin.12682

[37] A. Nazir, J. He, N. Zhu, S.S. Qureshi, S.U. Qureshi, F. Ullah, et.al. “A deep learning based novel hybrid CNN-LSTM architecture for efficient detection of threats in the IoT ecosystem,” Ain Shams Engineering Journal, 15(19): 102777, 2024. doi: 10.1016/j.asej.2024.102777

[38] A.K. Tyagi, S. Kumari, S. Tiwari, “Internet of Things for smart healthcare: A survey,” Future of AI in medical imaging IGI Global Scientific Publishing, 19-41, 2024. doi: 10.4018/979-8-3693-2359-5.ch002

[39] A. Bibi, J. Sami Ur Rahman, “Machine Learning Enabled In-Home ECG: A Review,” Medinformatics, 1–14, 2025. doi: 10.47852/bonviewMEDIN42024336

Appendix 1: Python Program for Designing a Deep Neural Network to Predict the Status of Cardiac Patients by Emergency Technicians.

```
# Required libraries
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import classification_report
import tensorflow as tf
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input, Conv1D, MaxPooling1D, BatchNormalization, ReLU
from tensorflow.keras.layers import Bidirectional, LSTM, Dense, Dropout
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.optimizers import Adam

# Path to the Excel file containing the data (must include signal_0 to signal_4999 and two label columns)
file_path = "DATA.xlsx"
df = pd.read_excel(file_path)

# Separate features (signal_0 to signal_4999) and labels
X = df[[f'signal_{i}' for i in range(5000)]].values
y_screening = df['label_screening'].values
y_disease = df['label_disease'].fillna("None").values  # For non-critical cases, the value is None

# Normalize ECG signals using the Z-score method
X = (X - X.mean(axis=1, keepdims=True)) / X.std(axis=1, keepdims=True)

# Reshape to (samples, signal_length, 1) for CNN input
X = X.reshape(-1, 5000, 1)

# Encode labels with LabelEncoder and then One-Hot
screening_encoder = LabelEncoder()
y_screening_encoded = screening_encoder.fit_transform(y_screening)
y_screening_encoded = tf.keras.utils.to_categorical(y_screening_encoded, num_classes=3)

disease_encoder = LabelEncoder()
y_disease_encoded = disease_encoder.fit_transform(y_disease)
y_disease_encoded = tf.keras.utils.to_categorical(y_disease_encoded, num_classes=13)  # 12 diseases + None

# Split data into training, validation, test (70/15/15)
X_train, X_temp, y_scr_train, y_scr_temp, y_dis_train, y_dis_temp = train_test_split(
    X, y_screening_encoded, y_disease_encoded, test_size=0.3, random_state=42)
X_val, X_test, y_scr_val, y_scr_test, y_dis_val, y_dis_test = train_test_split(
    X_temp, y_scr_temp, y_dis_temp, test_size=0.5, random_state=42)

# Add light Gaussian noise to training data (data augmentation)
def add_gaussian_noise(X, std=0.01):
    noise = np.random.normal(0, std, X.shape)
    return X + noise

X_train_aug = add_gaussian_noise(X_train)

# Define CNN + BiLSTM architecture as per the article
input_layer = Input(shape=(5000, 1))

# First convolutional layer
x = Conv1D(32, kernel_size=5, padding='same')(input_layer)
x = BatchNormalization()(x)
x = ReLU()(x)
x = MaxPooling1D(pool_size=2)(x)

# Second convolutional layer
x = Conv1D(64, kernel_size=5, padding='same')(x)
x = BatchNormalization()(x)
x = ReLU()(x)
x = MaxPooling1D(pool_size=2)(x)

# First and second BiLSTM layers
x = Bidirectional(LSTM(64, return_sequences=True))(x)
x = Bidirectional(LSTM(32))(x)

# Final Dense layer before output
x = Dense(128, activation='relu')(x)
x = Dropout(0.5)(x)

# First output: Three-level screening
output_screening = Dense(3, activation='softmax', name='screening_output')(x)

# Second output: Disease type only in Critical state (13 classes including None)
output_disease = Dense(13, activation='softmax', name='disease_output')(x)

# Define multi-task model
model = Model(inputs=input_layer, outputs=[output_screening, output_disease])

# Compile model with Adam optimizer and cross-entropy loss for each output
model.compile(
    optimizer=Adam(learning_rate=0.001),
    loss={
        'screening_output': 'categorical_crossentropy',
```

```
        'disease_output': 'categorical_crossentropy'
    },
    metrics={
        'screening_output': 'accuracy',
        'disease_output': 'accuracy'
    }
)

# Define EarlyStopping to stop training after 10 epochs without improvement
early_stop = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights=True)

# Train the model
history = model.fit(
    X_train_aug,
    {'screening_output': y_scr_train, 'disease_output': y_dis_train},
    validation_data=(X_val, {'screening_output': y_scr_val, 'disease_output': y_dis_val}),
    epochs=100,
    batch_size=32,
    callbacks=[early_stop],
    verbose=2
)

# Evaluate model performance on test set
y_pred_scr, y_pred_dis = model.predict(X_test)

# Convert predictions back to text labels
y_scr_true = np.argmax(y_scr_test, axis=1)
y_scr_pred = np.argmax(y_pred_scr, axis=1)
y_dis_true = np.argmax(y_dis_test, axis=1)
y_dis_pred = np.argmax(y_pred_dis, axis=1)

# Screening evaluation report
print("📊 Screening Report:")
print(classification_report(y_scr_true, y_scr_pred, target_names=screening_encoder.classes_))

# Disease type evaluation report (only in Critical state)
print("📊 Disease Type Report:")
print(classification_report(y_dis_true, y_dis_pred, target_names=disease_encoder.classes_))
```

Appendix 2: Python program for designing a neural network for intelligent screening of cardiac patients based on the input of 9 manually extracted parameters from each ECG.

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import classification_report

# 1. Read data from an Excel or CSV file
df = pd.read_excel("features_data.xlsx")  # File must include 9 features + label

# 2. Define mappings from text to numeric
mapping_rhythm = {'regular': 0, 'irregular': 1}
mapping_st = {'normal': 0, 'elevated': 1, 'depressed': -1}
mapping_t = {'normal': 0, 'inverted': 1, 'peaked': 2}
mapping_axis = {'normal': 0, 'left_deviated': -1, 'right_deviated': 1}
mapping_p = {'normal': 0, 'wide': 1, 'absent': 2}

# 3. Convert text features to numeric
df['heart_rhythm'] = df['heart_rhythm'].map(mapping_rhythm)
df['ST'] = df['ST'].map(mapping_st)
df['T_wave'] = df['T_wave'].map(mapping_t)
df['cardiac_axis'] = df['cardiac_axis'].map(mapping_axis)
df['P_wave'] = df['P_wave'].map(mapping_p)

# 4. Separate features and label (initial diagnosis: Normal / Borderline / Critical)
X = df[['heart_rhythm', 'heart_rate', 'PR_interval', 'QRS', 'ST', 'T_wave', 'QT', 'cardiac_axis', 'P_wave']]
y = df['initial_diagnosis']  # Assumption: This column is the label

# 5. Encode labels
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)

# 6. Normalize features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# 7. Split the data into training and test
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y_encoded, test_size=0.2, random_state=42)

# 8. Define and train an MLP model
model = MLPClassifier(hidden_layer_sizes=(32, 16), activation='relu', solver='adam', max_iter=300, random_state=42)
model.fit(X_train, y_train)

# 9. Evaluate model
y_pred = model.predict(X_test)
print("📊 Model Performance Report:")
print(classification_report(y_test, y_pred, target_names=label_encoder.classes_))
```

Usage for Predicting a New Patient (Real Example)

python

```
# Features of the new patient in text form
new_patient = {
    'heart_rhythm': 'irregular',
    'heart_rate': 110,
    'PR_interval': 160,
    'QRS': 120,
    'ST': 'elevated',
    'T_wave': 'inverted',
    'QT': 420,
    'cardiac_axis': 'left_deviated',
    'P_wave': 'absent'
}

# Convert to numeric and DataFrame
row = pd.DataFrame([{
    'heart_rhythm': mapping_rhythm[new_patient['heart_rhythm']],
    'heart_rate': new_patient['heart_rate'],
    'PR_interval': new_patient['PR_interval'],
    'QRS': new_patient['QRS'],
    'ST': mapping_st[new_patient['ST']],
    'T_wave': mapping_t[new_patient['T_wave']],
    'QT': new_patient['QT'],
    'cardiac_axis': mapping_axis[new_patient['cardiac_axis']],
    'P_wave': mapping_p[new_patient['P_wave']]
}])

# Normalize features
row_scaled = scaler.transform(row)

# Predict class
pred = model.predict(row_scaled)
pred_label = label_encoder.inverse_transform(pred)
print("🔍 Initial Diagnosis for New Patient:", pred_label[0])
```

Assumptions*:* For each patient, the ECG is recorded according to the specifications emphasized in the article, and the 9 highlighted features are documented.

Appendix 3: Comparison Table of the Manual Extraction Approach for 9 Parameters and the Use of Raw 10-Second 5000-Point Signal.

Aspect	Manual Extraction of 9 Parameters	Raw 10-Second 5000-Point Signal
Data Input	9 manually extracted ECG features (heart rhythm, heart rate, PR interval, QRS, ST, T wave, QT, cardiac axis, P wave)	Raw ECG signal with 5000 data points over 10 seconds
Preprocessing	Conversion of text-based features to numeric values, followed by standardization (Z-score)	Z-score normalization and reshaping to (samples, 5000, 1) for CNN input
Model Architecture	MLP (Multi-Layer Perceptron) with two hidden layers (32, 16) neurons	CNN + BiLSTM with two convolutional layers and two bidirectional LSTM layers
Output	Initial diagnosis: Normal, Borderline, Critical	Two outputs: Screening (Normal, Borderline, Critical) and Disease Type (13 classes, including None)
Training Complexity	Lower complexity due to fewer input dimensions (9 features)	Higher complexity due to large input size (5000 points) and deeper network
Data Augmentation	Not applied	Light Gaussian noise was added to the training data
Training Time	Shorter due to a simpler model and fewer inputs	Longer due to complex architecture and large input size
Performance Metrics	Evaluated using classification report (precision, recall, F1-score)	Evaluated using a classification report for both screening and disease type outputs
Use Case	Suitable for environments with limited computational resources or manual feature extraction	Suitable for automated, high-precision analysis with access to raw ECG signals
Advantages	Simpler implementation, lower computational requirements, interpretable features	Captures more detailed patterns in the raw signal, potentially with higher accuracy
Limitations	Relies on manual feature extraction, may miss subtle patterns in the raw signal	Requires significant computational resources, less interpretable

10.57647/ijm2c.2025.1504.20