

An Intelligent Low-Cost Digital Stethoscope Using Arduino, Signal Processing, and Machine Learning for Real-Time Heart Sound Analysis

Nafisa M. Mapari,¹ Alefiya J. Managori,^{2*} Hajra A. Naik,³ Monika G. Chaurasiya⁴ and Tasmiya I. Kandhal⁵

Department of Computer Engineering, Anjuman-I-Islam M. H. Saboo Siddik College of Engineering, Mumbai, Maharashtra, 400008, India

*Corresponding Author

Alefiya J. Managori

✉ alefiya.231233.co@mhssce.ac.in

Received: 20 April 2026

Revised: 22 May 2026

Accepted: 25 May 2026

Published Online: 03 June 2026.

Citation

N. M. Mapari, A. J. Managori, H. A. Naik, M. G. Chaurasiya, T. I. Kandhal, An intelligent low-cost digital stethoscope using Arduino, signal processing, and machine learning for real-time heart sound analysis, *Journal of Smart Sensors and Computing*, 2026, 2(2), 26206, <https://doi.org/10.64189/ssc.26206>.

Open Access

This article is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/), which permits the non-commercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as appropriate credit to the original author(s) and the source is given by providing a link to the Creative Commons License and changes need to be indicated if there are any. The images or other third-party material in this article are included in the article's Creative Commons License, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons License and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this License, visit: <https://creativecommons.org/licenses/by-nc/4.0/>

© The Author(s) 2026

Abstract

Cardiovascular diseases remain among the leading causes of mortality worldwide, creating a growing need for affordable and accessible cardiac monitoring systems. Traditional acoustic stethoscopes rely heavily on physician expertise and do not provide capabilities such as signal visualization, storage, or automated analysis. To address these limitations, this paper presents an intelligent low-cost digital stethoscope system using Arduino, signal processing, and machine learning for real-time heart sound analysis. The proposed system uses an Arduino UNO microcontroller integrated with a modified KY-037 heart sound acquisition module and an external electret microphone coupled with a stethoscope chest piece for phonocardiogram acquisition. The acquired signals are transmitted through USB serial communication and processed using a multistage signal preprocessing pipeline consisting of median filtering, Butterworth bandpass filtering, moving average smoothing, and outlier suppression to improve signal quality and reduce environmental noise. The processed heart sound signals are analyzed for real-time heart rate estimation using adaptive peak detection techniques. In addition, a machine learning-based cardiac sound classification module is implemented using a random forest classifier trained on the PhysioNet CirCor DigiScope phonocardiogram dataset. Feature extraction is performed using Mel Frequency Cepstral Coefficients (MFCCs), spectral features, chroma features, and signal energy parameters. The developed system also includes a Streamlit-based monitoring dashboard for live waveform visualization, BPM tracking, patient record management, data storage, CSV export, and PDF report generation. Experimental evaluation demonstrated successful real-time heart sound acquisition and an overall classification accuracy of 82%, with effective waveform visualization and stable BPM estimation. The proposed system provides a compact, economical, and scalable solution for intelligent cardiac monitoring and has potential applications in low-resource healthcare environments, telemedicine, and biomedical education.

Keywords: Digital stethoscope, Arduino UNO, Phonocardiogram, Signal processing, Machine learning, Random Forest classifier, Cardiac monitoring.

1. Introduction

Cardiovascular disease (CVD) is currently the number one cause of morbidity and mortality worldwide and is thus a significant concern for all global healthcare systems. Early and ongoing detection of cardiovascular conditions can help prevent morbidity and mortality due to cardiovascular complications. The initial diagnosis of CVD has historically been achieved through an auscultative approach involving the use of noninvasive techniques (i.e., using a stethoscope). The primary challenge with the use of a stethoscope is the need for an appropriate skill level and experience of the clinician to provide an accurate diagnosis and the lack of ability to visually display (display), record/store (store) and automate (automate) analyses of heart sounds from the stethoscope.^[1] If a clinician is unable to accurately diagnose cardiovascular disease using a stethoscope, other diagnostic testing methods (other than a stethoscope) may lead to additional problems with respect to detecting and diagnosing accurate cardiovascular disease.^[2]

New developments in both biomedical instrumentation and Intel-based systems have led to new types of digital stethoscopes that can provide a greater ability to acquire and analyze heart sounds.^[2,3] A digital stethoscope converts the acoustic cardiac signal produced by the heart into an electrical signal that is capable of being processed, displayed, and stored using standard computer-based techniques. Many studies have used microphone modules, embedded digital computers, and other signal conditioning methods to acquire phonocardiograms and monitor heart patients.^[3,4] Additionally, the collaboration between universities and private/nonprofit organizations has generated significant recognition concerning new types of low-cost wearable medical devices that are based on open-source hardware platforms because of their low cost, small size/weight, and ease of use.^[2,5] Therefore, advancements in digital medical technology have created new opportunities for new types of intelligent and affordable healthcare systems.

In recent years, many advancements have been made in how we investigate and analyze certain biological signals, such as heart sounds, using signal processing and machine learning methods. Many different methods have been used to filter and remove noise from recordings of heart sounds (phonocardiograms) so that better quality recordings can be obtained and that noise introduced by the external environment can be reduced.^[6,7] Recent work has demonstrated that machine learning and deep learning techniques show promise for the automatic classification of heart sounds, as well as for the detection of abnormalities.^[3,8,9] The availability of numerous publicly available biological datasets, such as PhysioNet and the CirCor DigiScope database, has allowed

researchers to obtain standardized recordings of these heart sounds to develop and test algorithms for intelligent cardiac monitoring systems.^[10,11,12] Additionally, researchers are using advanced software frameworks and visualization tools to create applications to monitor patients in real time, manage patient data remotely and develop interactive dashboards for healthcare providers to enhance healthcare services.^[2,13]

As cost-effective smart medical devices have been developed to meet the growing demand for such devices, this project presents a new low-cost digital stethoscope system that utilizes an Arduino microcontroller and digital signal processing (DSP) and machine learning (ML) algorithms to analyze heart sounds. The heart sound acquisition system is based on the modified KY-037, which includes an external electret microphone for collecting a PCG (phonocardiograms) using an Arduino UNO microcontroller. The retrieved PCG signals undergo a detailed, multistage preprocessing pipeline that includes a median filter (MF), Butterworth bandpass filter (BPF), moving average filter (MAF), and outlier removal to create quality heart sounds. Heart sounds are classified by using the random forest algorithm through a previously developed ML classifier, which uses the PhysioNet CirCor DigiScope database for training data collection.^[11] The real-time PCG data collected are displayed as a visual waveform on a Streamlit-based dashboard, along with additional functionality such as BPM monitoring, patient record management, data storage and automated PDF report generation. Overall, this proposed architecture provides a low-cost, compact, and portable solution for academic, research, and initial patient screening purposes.

This study contributes greatly to the development of a low-cost intelligent digital stethoscope through the use of open-source hardware/software technologies, developing a multistage heart sound preprocessing pipeline that minimizes noise and improves the quality of a signal, providing a real-time estimate of the BPM through an adaptive peak detection method, and classifying heart sounds using machine learning and the PhysioNet database.^[11,12] Additionally, the inclusion of a Streamlit monitoring Dashboard, which includes features such as waveform visualization, patient record management, and automated PDF report generation, creates additional capabilities for future smart healthcare solutions and allows for further scaling.^[2,13]

2. Methodology

A new intelligent digital stethoscope system has been produced using a combination of embedded hardware, signal processing methods, machine learning algorithms, and visualization tools based on software for monitoring and analyzing sounds produced by the heart. The design approach is based on using low-cost methods for

recording heart sounds, processing these data to make BPM estimates, using intelligent techniques for classifying sounds, and providing interactive methods for monitoring patients through the use of open-source solutions.

The architecture of the system consists of an acquisition module made from an Arduino UNO board connected to a modified KY-037 microphone and an external electret microphone for recording phonocardiograms. The recorded signals are processed using multiple filtering techniques and analyzed with machine learning techniques that have been trained with the PhysioNet CirCor DigiScope dataset. A Streamlit-based dashboard has also been developed to provide real-time visualization of the waveforms and to monitor the BPM, manage patient information, and create reports automatically. The detailed methodology and stages of the implementation of the proposed system are discussed in the subsequent subsections.

2.1 System overview

The intended intelligent digital stethoscope system was developed to enable heart sound acquisition in real time, signal preprocessing, BPM (beats per minute) estimation, machine-learning-based classification, and interactive monitoring via a software interface or dashboard. Initially, the heart sounds are captured using an external electret microphone attached to the stethoscope chest piece and connected to a modified KY-037 sound acquisition module. The Arduino UNO microcontroller acquires the analog signals via its analog input pins, and these analog signals are converted to digital signals using the built-in (on-board) ADC (analog-to-digital converter). Once the signals have been acquired, they are transmitted via USB serial communication to a PC for further signal processing and analysis. A multistage preprocessing pipeline consisting of median filtering, Butterworth bandpass filtering, moving average smoothing, and outlier suppression is implemented to improve signal quality and reduce environmental noise levels. After processing, the heart rate is estimated in the BPM via adaptive peak detection methods.

The classification of processed heart sound signals are accomplished through the implementation of a machine learning model trained on PhysioNet CirCor DigiScope phonocardiogram data. The model predicts whether the acquired heart sound can be classified as either a normal or murmur. The processed heart sound signals, BPM, prediction results, and patient information will be made available through an application built on the Streamlit web framework, which will allow for visual representation of the waveforms, storage in a database, export to CSV format, and report generation in PDF format. The complete workflow of this proposed intelligent digital stethoscope is illustrated in the graphical abstract.

2.2 Working principle

The intelligent digital stethoscope system operates as follows: the heart sound sample is collected in real time and then preprocessed, the BPM is estimated, and the data are classified via machine learning and then visualized on a dashboard. The heart sounds collected are converted to a digital signal via an Arduino UNO microcontroller and then passed through a series of filters to reduce noise and enhance the waveform. Next, the processed signals are used for estimating BPM and for classifying the cardiac sounds using machine learning. Finally, the results are visualized on a Streamlit-based dashboard with data storage and reporting capabilities.

2.2.1 Hardware

The proposed hardware architecture consists of an Arduino UNO microcontroller, a modified KY-037 sound acquisition module, and one or more of the following peripherals: an external electret microphone(s) and a stethoscope chest piece(s). The prototype hardware is connected to a compatible computer via USB communication. A block diagram of the proposed prototype is shown in Fig. 1, while a picture of the experimental hardware setup is shown in Fig. 2.

The Arduino UNO microcontroller serves as both the central processing unit and the data acquisition unit for the proposed system. The architecture of the Arduino

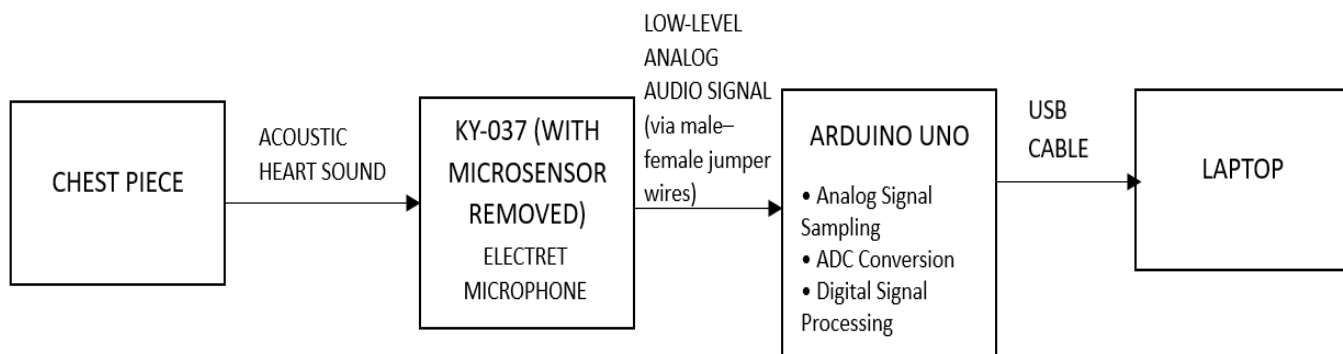


Fig. 1: Block diagram of the proposed Arduino-based digital stethoscope prototype.

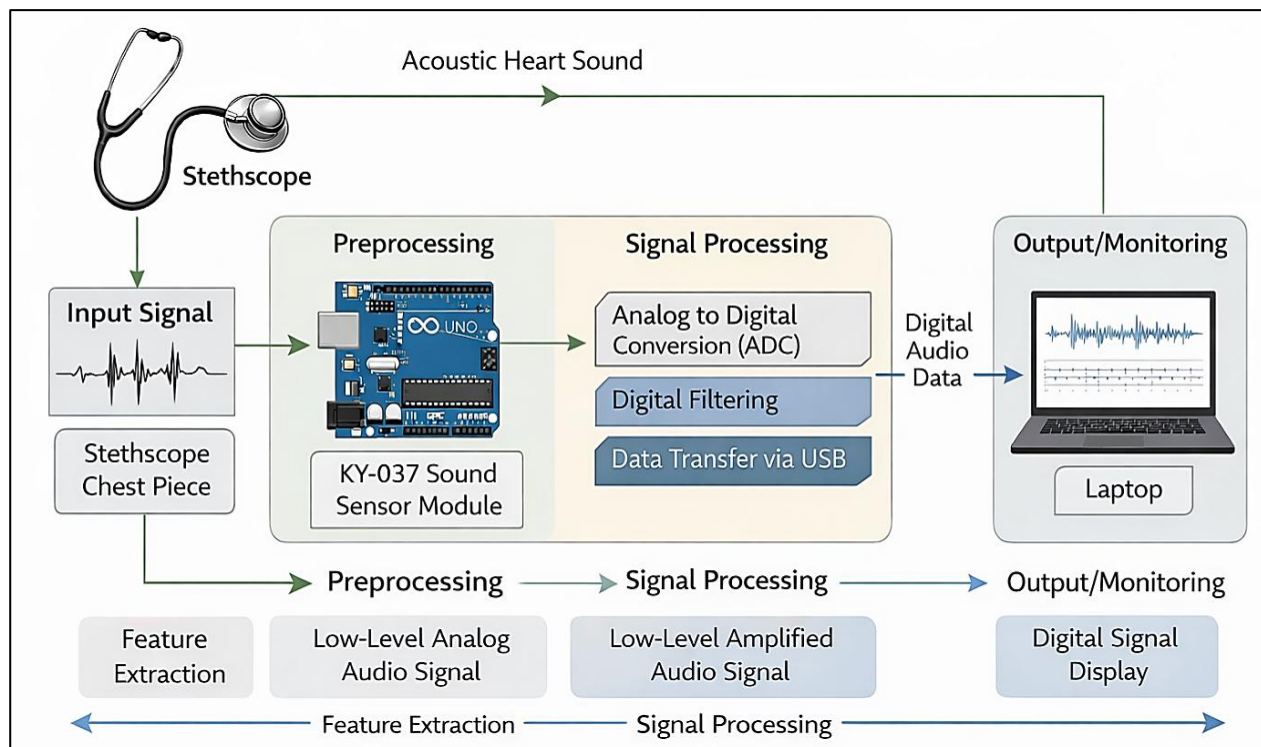


Fig. 2: Experimental hardware setup of the developed digital stethoscope system.

UNO uses ATmega328P and has multiple analog input channels to acquire real-time biomedical signals. The analog heart sound signal is converted to a digital value ranging from 0 to 1023 by the integrated 10-bit analog-to-digital converter (ADC) within the Arduino.^[14] After acquiring the data, Arduino processes it (smoothing) and communicates with the computer over a serial communication link using a baud rate of 9600 bps.

Sound will be captured with a customized KY-037 sound sensor, with the original microphone located onboard KY-037 removed and an external electret microphone incorporated into a pair of stereo earphones. To provide better acoustic coupling between the heart and microphone and increase the sensitivity to low-amplitude heart sound signal events, the new microphone was added internally to the stethoscope chest piece. The stethoscope chest piece will create a concentrated area in which acoustic vibrations produced by cardiac activity will have an outlet to transmit sound, resulting in a better quality of sounds being received.^[4] The signals received from the heart are transmitted to a laptop system via a USB connection with no requirement for additional power, and the laptop system is used as the main processing and visualization platform where signal preprocessing, BPM estimation, machine learning inference and dashboard visualization are accomplished using Python-based software frameworks. The overall architecture is aligned with the vision of an inexpensive intelligent digital stethoscope as well as recently proposed IoMT-based healthcare monitoring systems in the surgical realm.^[13]

2.2.2 Signal acquisition and preprocessing

Heart sounds were recorded using both a modified KY-037 sound sensor module and an external electret microphone through a stethoscope chest piece. The heart-generated acoustic vibrations were picked up by the external microphone and then converted to analog electrical signals. The analog signals were sent to the A0 analog input pin on the Arduino UNO for digitization and subsequent processing. Since heart sounds are low-amplitude biomedical signals, careful acquisition and preprocessing are required to reduce the amount of environmental interference, motion artifacts, and signal instability. Studies involving phonocardiogram (PCG) analysis have suggested that signal enhancement and denoising techniques should be used to improve the quality and reliability of acquired cardiac sounds.^[6,7]

The Arduino UNO uses its onboard 10-bit analog-to-digital converter (ADC) to convert the analog heart sound signal into digital values ranging from 0 to 1023. The ADC conversion process can be represented as follows:

$$V_{in} = \frac{ADC \times V_{ref}}{1023} \quad (1)$$

where V_{in} represents the input analog voltage, ADC denotes the digital ADC value, and V_{ref} corresponds to the reference voltage of the microcontroller.

The sampling frequency of the system was maintained at approximately 100 Hz using a 10 ms delay between consecutive samples. The sampling frequency is calculated as follows:

$$f_s = \frac{1}{T_s} \quad (2)$$

where f_s represents the sampling frequency and T_s denotes the sampling interval.

The Arduino microcontroller carried out an 8-sample moving average smoothing (M.A.S.) directly on the device to smooth out fluctuations and make the signal more stable at the point of acquisition prior to transmitting it to the computer via a serial connection. The smoothed signal and BPM values were sent to the computer through a USB-based serial connection in a signal/BPM data format. The smoothed signal and BPM values were then continuously received with PySerial using a separate thread to allow for uninterrupted real-time signal processing and visualization.

2.2.3 Signal processing pipeline

Environmental noise, motion artifacts, electrical interference, and sudden spikes in heart sound signals may all impact acquired heart sound recordings. A multistage pipeline for preprocessing signals was developed to enhance the basic characteristics of low-cost electret microphones and the effects of external acoustic disturbances to provide reliable measurements of BPMs and accurate classifications based on machine learning. Similar signal preprocessing techniques, denoising, and segmentation techniques have been used in the analysis of phonocardiograms to enhance the quality of heart sound signals and improve the ability to interpret heart sounds for diagnostic purposes.^[6,4] The workflow of the signal-processing pipeline is shown in Fig. 3.

2.2.3.1 Median filtering

The first step of signal processing involved applying a median filter to the input signal to reduce the influence of sudden impact noise and transient spikes. A median filter with a kernel size of nine points was used to smooth out the acquired waveform. As median filtering is a nonlinear filtering operation, it was able to remove singular outlier points while maintaining the basic morphology of the heart sound waveform.^[4,6] The application of median filtering increased the stability of the signal and reduced sudden amplitude variations caused by microphone handling and environmental disturbances.

2.2.3.2 Butterworth bandpass filtering

After the median filtering step was complete, the signal was filtered using a fourth-order Butterworth bandpass filter with cutoff frequencies of 0.5 Hz (low) and 15.0 Hz (high). Low-frequency completion of the Butterworth filter removed baseline drift and low-frequency motion artifacts, whereas high-frequency completion of the Butterworth filter removed high-frequency electrical

noise and unwanted acoustic interference from the signal. The use of the `scipy.signal.butter` and `filtfilt` functions allowed zero-phase filtering of the signal without distorting its waveform. Butterworth filtering techniques are very common methods for enhancing the phonocardiogram signal and removing noise from biomedical signals to enhance the clarity and reliability of the signal.^[7]

2.2.3.3 Moving average smoothing

Once the signal was bandpass filtered, a smoothing technique that uses a moving average filter was applied to lessen the minor fluctuations and any residual ripples still present in the signal. The moving average filter has made the cardiac waveform smoother, thus resulting in better consistency between cardiac cycles that have occurred sequentially. This means that after being smoothed, the stability of the processed data is improved when the beats per minute are estimated, thus increasing the reliability of the extracted features and providing a better basis for machine learning analysis of the data collected.

2.2.3.4 Outlier clamping

During the last stage of preprocessing, outlier clamping was applied to the signal data to eliminate any remaining extreme peaks or spikes prior to performing the waveform analysis. By clamping any values that exceed ± 3 standard deviations from the original mean amplitude, this operation suppresses transitory artifacts and sudden peak conditions. This will also increase the consistency of the cardiac waveform and significantly reduce false positive detections of peak values when the beats per minute are estimated. Other types of acoustic signal processing noise reduction techniques have also been reported to provide a more reliable means of processing phonocardiograms through various methods of signal stabilization.^[6,7]

2.2.4 Heart rate (BPM) estimation

Techniques for adaptive peak detection were applied to analyze heart sounds that were filtered to determine the heart rate of the patient (BPM). To estimate the BPM, the envelope characteristics of the waveform were filtered, and each peak point was identified as a corresponding point of a cardiac cycle. Methods for the segmentation of heart sounds and passing through the peaks as described above have routinely been used within the analysis of phonocardiograms (PCGs) to reliably detect cardiac events.^[4,6] The peaks in the processed heart sound signal were automatically found using the functionality available through the SciPy library to find peaks within the dataset.

To improve the detection accuracy, an adaptive prominence threshold equal to 0.5 times the standard

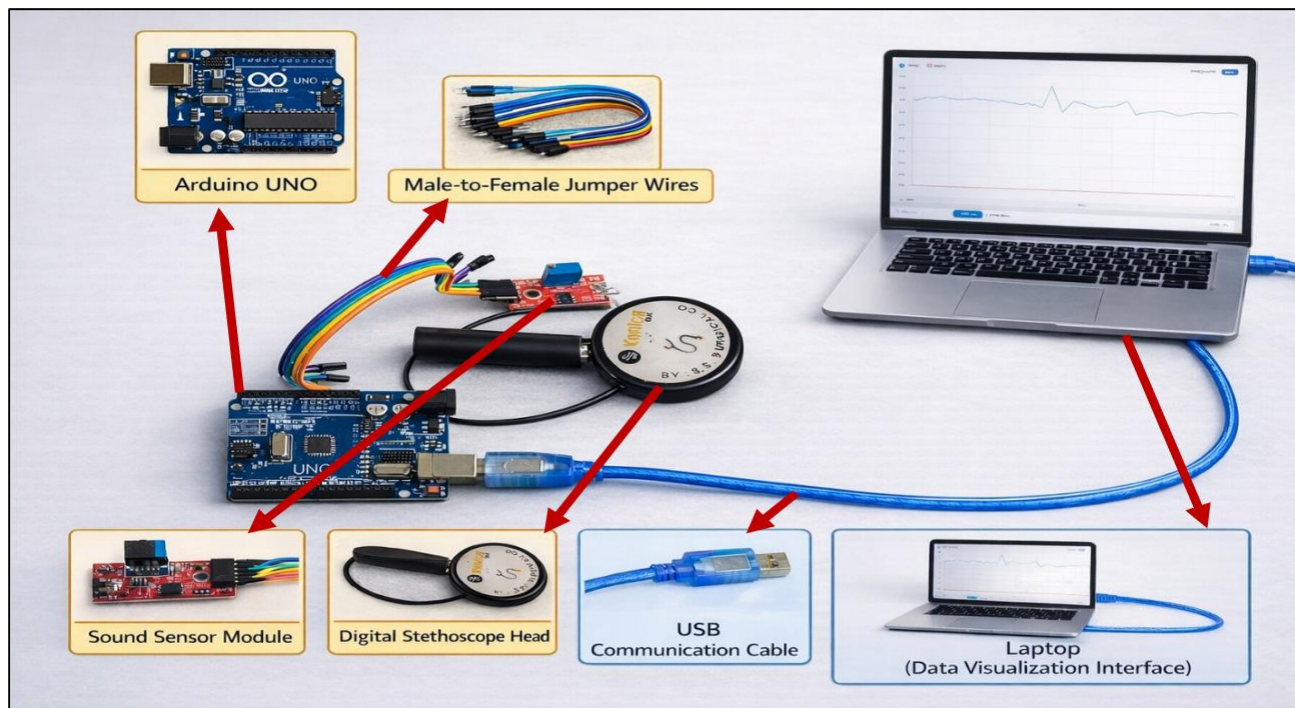


Fig. 3: Schematic overview of the proposed heart sound signal processing pipeline.

deviation of the processed signal was employed. In addition, a minimum peak distance of 0.35 seconds was maintained to avoid false detections and support heart rate measurements up to approximately 170 BPM. The BPM value was calculated using the mean interpeak interval as follows:

$$BPM = \frac{60}{\text{Mean Inter-Peak Interval}} \quad (3)$$

The estimated BPM values were validated within a physiological range of 40–180 BPM to eliminate unrealistic measurements caused by noise or motion artifacts. The BPM results were continuously updated and visualized in real time through the Streamlit monitoring dashboard.

2.2.5 Machine learning-based classification

2.2.5.1 Dataset description

A machine learning classification model was developed and trained using the PhysioNet CirCor DigiScope phonocardiogram dataset.^[11,12] The CirCor DigiScope database consists of 1507 heart sounds from 474 patients; there are 2 major classes made up of recorded heart sounds, such as “normal/healthy” heart sounds and “abnormal/murmur” heart sounds. Recordings have been validated clinically and annotated by physician specialists, who have created a viable database for melodic cardiac sound analysis and machine learning-based verification of heart sound categories.

2.2.5.2 Data balancing

In the original database, there is a significant imbalance between the number of normal and murmur records

present in the dataset, with a predominance of the normal category of recordings. To mitigate bias during training toward the majority class, random oversampling was performed on the murmur class of recordings until an equal number of records from both classes were represented in both the training and evaluation phases of machine learning production, thus increasing classification fairness and reducing sensitivity to abnormal heart sounds.

2.2.5.3 Feature extraction

Using the Librosa audio analysis library,^[15] clinicians were able to extract Mel Frequency Cepstral Coefficients (MFCCs) from their heart sound recordings. MFCCs are widely accepted for their ability to represent both temporal and spectral aspects of the phonocardiogram signal. They are therefore commonly used in biomedical audio analysis and in heart sound classification systems.^[7] In addition to MFCCs, clinicians extracted additional complementary acoustic features that included spectral centroid, spectral bandwidth, spectral roll off, zero-crossing rate (ZCR), root mean square (RMS) amplitude, and chroma features. A total of 77 features were extracted per recording for subsequent machine learning analysis, as shown in [Table 1](#).

2.2.5.4 Random forest classification

Random Forest classifier was chosen to classify heart sounds because it is adaptable, strong, simple to apply, and works well for processing high-dimensional biomedical features.^[16] Many current studies have shown how machine learning and deep learning methods can be

Table 1: Feature extraction summary used for heart sound classification.

Feature Type	Number of Features	Purpose
MFCC Mean	20	Captures tonal characteristics of heart sounds
MFCC Standard Deviation	20	Represents variability in heart sound patterns
MFCC Delta Mean	20	Captures temporal variations in cardiac signals
Spectral Centroid	1	Represents center frequency distribution
Spectral Bandwidth	1	Measures frequency spread
Spectral Rolloff	1	Indicates high-frequency energy distribution
Zero Crossing Rate	1	Measures signal irregularity
RMS Energy	1	Represents signal amplitude and energy
Chroma Features	12	Captures harmonic and tonal information
Total Features	77	Combined feature vector used for classification

used for automatic murmur detection and intelligent analysis of heart sounds.^[3,8] Our model utilized 300 decision tree estimators with balanced weightings for all the associated classes to increase the sensitivity for detecting murmurs. Additionally, recent research in the field of evaluating biomedical signals has researched alternative methods for classifying heart sounds using intelligent phonocardiograms.^[9]

Model validation involved the use of fivefold stratified cross-validation to ensure that performance was evaluated on the basis of different partitions of data. The prediction threshold was decreased from its default value of 0.50 to a lower value of 0.30 to increase the sensitivity of murmur detection. During cross-validation, the proposed model achieved an overall classification accuracy of 82%, with a corresponding F1 score of 95.9%. Owing to the nature of our model, it could be integrated into existing real-time monitoring systems and thus provide for both the live extraction of heart sound data and the automatic detection and intelligent analysis of such sounds.

2.2.6 Streamlit-based monitoring dashboard

To facilitate real-time analysis of heart sounds and efficiently manage monitoring data for patients, an interactive dashboard has been developed using the Streamlit framework, as shown in Figs. 4 and 5. This dashboard includes a number of features, including but is not limited to, live scrolling waveform visualization, trends of beats per minute (BPM), confidence scores for machine learning predictions, and organized display of patient information. Through continuous and uninterrupted acquisition and display of heart sounds, this system offers the opportunity for healthcare providers to monitor their patients in real time. Other intelligent monitoring systems, such as IoMT-based healthcare applications, that offer real-time visualization of biomedical signals have been widely investigated for the purpose of physiological signal monitoring.^[2,13]

The waveform visualization component was developed using Matplotlib and functions at a refresh rate

of approximately 6 frames per second to provide smooth rendering of real-time signals. The BPM and machine learning prediction outputs were also updated continuously on dedicated metric panels to provide real-time feedback to the user. Prior to acquiring and analyzing the signals, patient name, age, identification (ID), and clinical notes can be entered by the user into the integrated patient information forms and displayed on the dashboard.

Additionally, the system allows for the continuous storage of executive records and historical monitoring data through an SQLite database. The functions of this component allow for long-term storage of historical data to provide for future clinical assessment. Usability and reporting were also enhanced by adding a means to export data to a CSV file and formatting automated PDF reports on the monitoring dashboard. The reports generated in Fig. 6 include waveform plots, BPM trends, heart sound classification results, confidence scores, and patient identification information. Integrating intelligent monitoring, data management, and reporting capabilities increases the potential use of the proposed system for biomedical research, educational purposes, and the smart healthcare monitoring industry.^[2]

2.3 Implementation

2.3.1 Real-time serial communication

The real-time communication between the Arduino UNO and the PC was via USB serial communication at 9600 Bps. The PySerial Library was used to receive continuous heart sound data sent from the microcontroller, and a background thread was created to provide nonblocking serial communication and uninterrupted real-time processing of the heart sound signals.

2.3.2 Live-signal visualization

Matplotlib integrated into the Streamlit Framework was used to visualize the acquired and processed heart sound signals in real time. The waveform plots were continuously updated with such updated waveforms; thus, live monitoring of cardiac activity could be achieved.

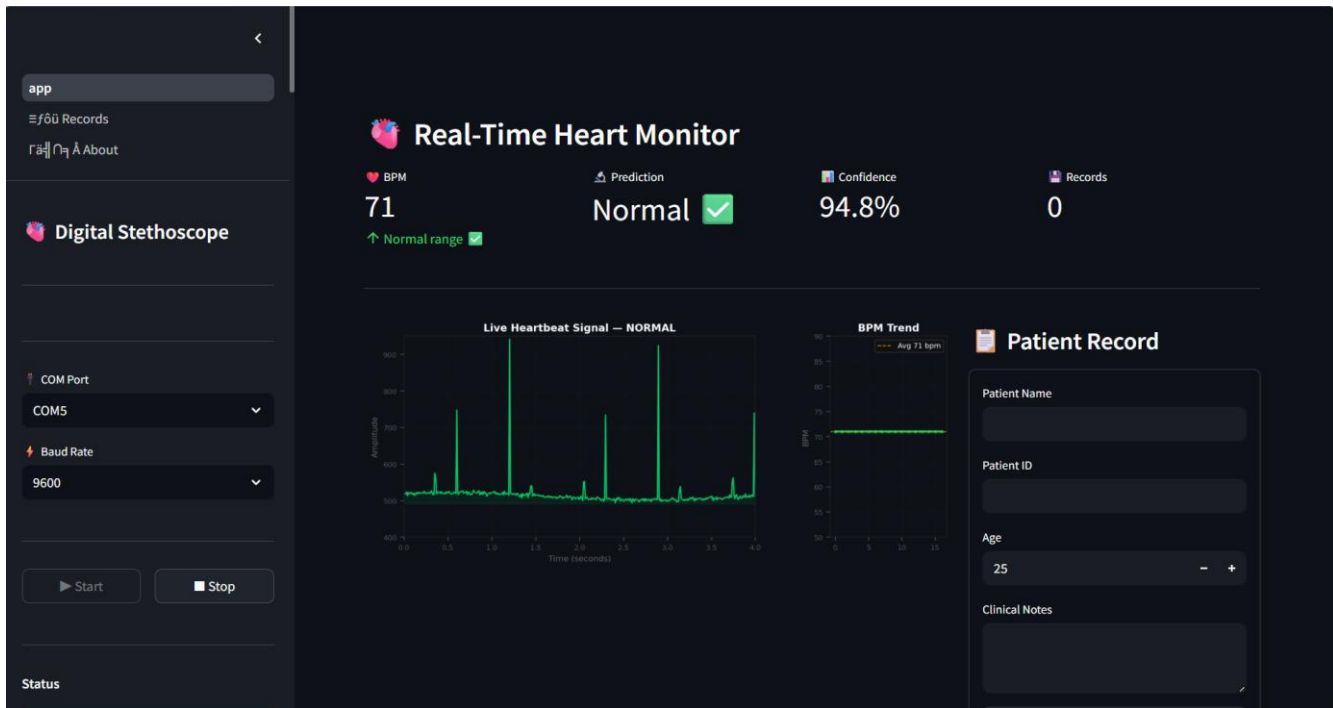


Fig. 4: Streamlit dashboard showing normal heart sound analysis and BPM monitoring.

In addition, the visualization module also provided the current BPM trends and prediction confidence values for interactive cardiovascular analysis.

2.3.3 Data storage and reporting

The patient records and monitoring output from the SQLite database were integrated into a Python application that stored the information in an SQLite database. The data were exportable to the CSV format (using the Pandas library) for further analysis and documentation. Automated PDF reports were generated via Matplotlib PdfPages, which included waveform plots, BPM trend analysis, patient information and machine learning predictions.

3. Results and analysis

3.1 Heart sound acquisition results

The validation experiment for the digital stethoscope

system was completed by using the Arduino IDE (serial plotter) to confirm the acquisition of sounds from the heart (heart sounds), as well as whether the system created waveforms of the heart sounds. After the sound waves were acquired using a modified KY-037 sensor module along with an external electret microphone (from the same acoustical source) and a stethoscope chest piece, the heart sound periodic (cyclical) signals captured from the subject were transmitted using USB serial communication and displayed in real time on the Arduino serial plotter interface.

The resultant raw waveform as output from the Arduino IDE incorporated recognizably cyclical (periodic) cardiac cycle representations of heartbeat (heart sound) activity. The heart sound cyclic report in the waveform output indicated proper functioning for both the proposed hardware equipment and the corresponding acquisition circuit to enable later

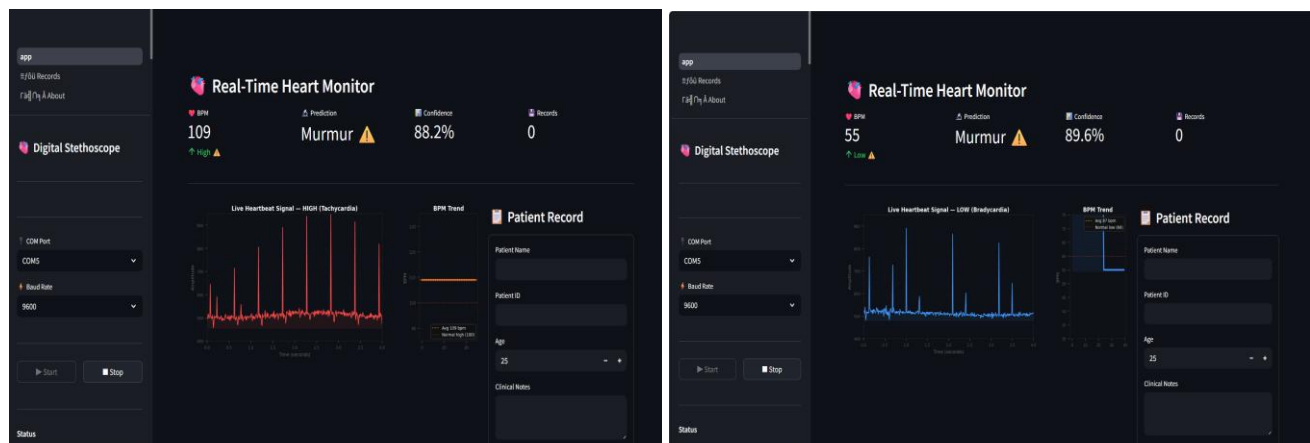


Fig. 5: Streamlit dashboard showing abnormal heart sound classification with murmur prediction.

Digital Stethoscope

Heart Sound Analysis Report

Generated: 10 April 2026, 10:08 AM

PATIENT INFORMATION

Patient Name: Monika
Age: 21 years
Time: 10:08 AM

Patient ID: 231205
Date: 10/04/2026
Device: Arduino + Earphone Mic

DIAGNOSIS RESULTS

Average BPM: 71 bpm
Min / Max BPM: 71 / 71 bpm
Confidence: 93.5%

BPM Status: **Normal**
ML Prediction: **Normal**
Classification: **Normal Sinus Rhythm**

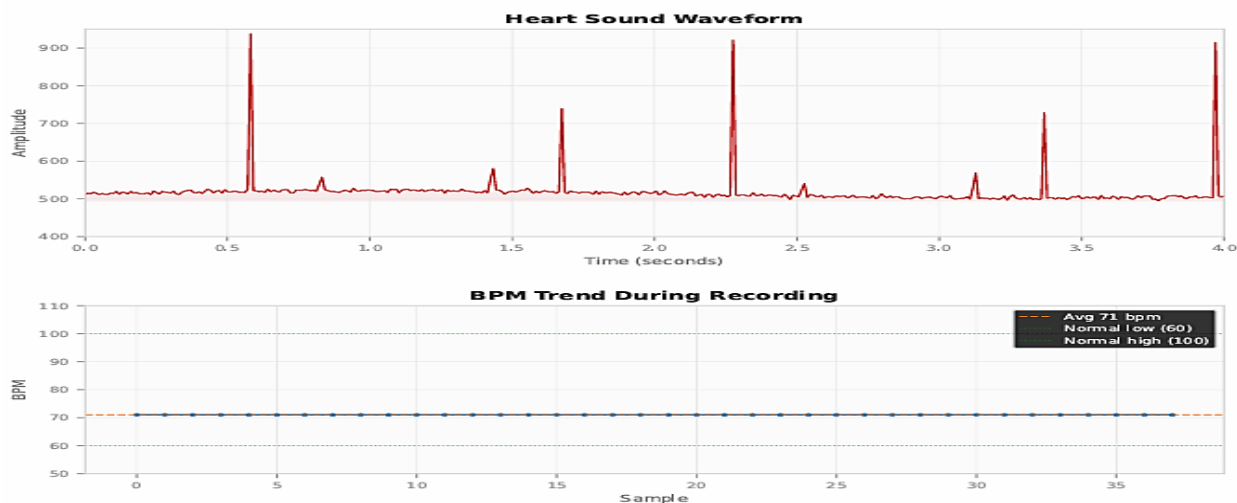


Fig. 6: Generated PDF-based heart sound analysis report produced by the proposed digital stethoscope system.

implementation of complex signal processing techniques and AI/machine learning algorithms. Although noise and some interference were evident in the raw signals, the peaks of the heart beats could be easily differentiated to allow subsequent processing and determination of heart rate in as many heart beats per minute (BPM).

After validation in the Arduino IDE, the acquired data were imported into a Streamlit dashboard developed in Python for further processing. The heart sound signals were pre-processed, visualized, and analyzed to provide real-time BPM estimation and machine learning-based cardiac sound classification. The real-time raw waveform obtained from the Arduino serial plotter is presented in Fig. 7.

3.2 Signal processing results

The multistage signal processing pipeline that was applied enabled the heart sound signals that were collected to have improved signal quality. The median filter was successful at eliminating sudden impulse-type noise and transient spikes caused by both motion of the microphone and other outside sources. The Butterworth bandpass filter was successful at eliminating baseline drifts in the low-frequency domain while also removing electrical noise from the high-frequency range, producing a more continuous, stable waveform pattern.

By using the moving average filter for additional smoothing of the waveform, waveform continuity was enhanced while small fluctuations in the signal were reduced. A final processing step for the signal was to clamp outliers, thus preventing large amplitude spikes that would have resulted in inaccurate strengths for future BPM calculations. The comparison between the processed heart sounds in real time and the raw heart sounds revealed clear differences in clarity and periodicity, indicating that the degree of filtering that was performed on the waveform was an excellent visual indicator.

The proposed steps described in the overall preprocessing pipeline were successful at improving the overall stability of the signal, resulting in greater confidence in the future estimation of BPM from the processed signal and greater reliability when machine learning-based classifications of heart sounds obtained from the same signal were performed. In addition, the filtering steps reduce the number of false peak detections because of motion artifacts or interference from nearby electronic sources (i.e., computers and machines).

3.3 BPM estimation, machine learning performance and comparative analysis

According to an experimental evaluation, the

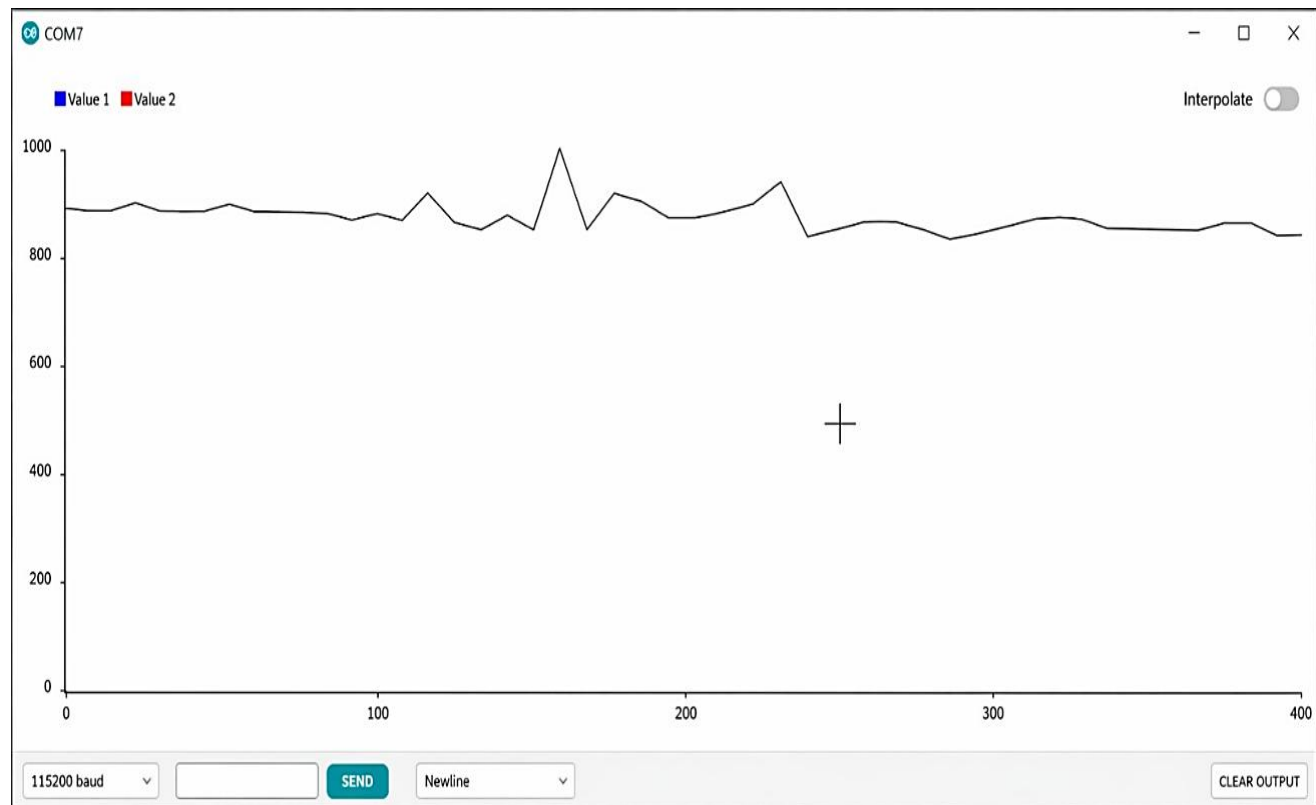


Fig. 7: Real-time raw heart sound waveform visualized using the Arduino IDE serial plotter.

intelligent digital stethoscope demonstrated stable real-time BPM estimation and effective machine learning-based cardiac sound classification. The adaptive peak detection algorithm was successful in detecting cardiac peaks from the processed phonocardiogram signals and generating BPM values in a physiological range of 40–180 beats per minute (BPM) on a continuous basis. The use of preprocessing stages greatly reduced false peak detection and improved the stability of the waveform during real-time monitoring. Finally, the developed system was able to monitor heart sounds continuously and display BPM values in real time via the Streamlit dashboard.

The random forest (RF) ML model trained using the PhysioNet CirCor DigiScope dataset identified normal heart sounds and murmur heart sounds. We successfully analyzed and classified cardiac sounds on the basis of the MFCC and the spectral, chroma, and energy features that we extracted. The overall accuracy of the classifier was measured at 82%, and the cross-validation F1 score was 95.9%. Normal heart sounds were classified with high recall; however, there was lower sensitivity for murmur classification because of discrepancies between how clinical grade systems record audio vs our low-cost electret microphone recording in the prototype setup. Real-time testing also affected performance when murmurs were distinguished because of environmental noise, sensitivity limitations of the microphone used, and inconsistencies in the chest piece placement path.

The KY-037-based digital stethoscope was compared with existing low-cost digital stethoscopes using the MAX9814 chip and with commercially available digital stethoscopes across a number of parameters, such as system cost, hardware architecture, monitoring capability, portability, and where they can be used, as shown in Table 2. The proposed system uses a cost-effective hardware architecture while providing real-time monitoring of cardiac signals, a software visualization program, and intelligent machine learning-based analysis. Compared with traditional low-cost digital stethoscopes, the proposed architecture can provide features such as multistage filtering of the signals, adaptive BPM estimation, storing of data in an SQLite database, exporting of the data in its CSV format, and automated PDF report generation via the Streamlit monitoring dashboard.

3.4 Discussion

The digital stethoscope system created successfully shows how a high-quality open-source platform could be created to allow for real-time acquisition of heart sound signals, to provide visual output of the acquired waveforms, to allow for preprocessing of the heart sounds, and to utilize intelligent analytical techniques to analyze the acquired heartbeat signal. All the experiments using the digital stethoscope produced consistent results when the modified KY-037 sensor module was used in conjunction with the Arduino UNO as the microcontroller to acquire heart sounds, with

Table 2: Comparative analysis of the proposed digital stethoscope system with existing implementations.

Parameter	Proposed System (KY-037 + Arduino)	MAX9814 Based System	Commercial Digital Stethoscope
Approximate Cost	₹1000–₹1500	₹1500–₹2500	₹15,000+
Microcontroller	Arduino UNO	Arduino/MCU	Proprietary Hardware
Sound Acquisition Module	Modified KY-037 with External Electret Microphone	MAX9814 (AGC Enabled)	Integrated Clinical Sensor
Real-Time Monitoring	Yes (Serial Plotter + Streamlit Dashboard)	Yes	Yes
Wireless Connectivity	No	Optional	Yes
Machine Learning Support	Yes	Limited	Proprietary
Data Storage and Reporting	SQLite, CSV Export, PDF Reports	Minimal	Integrated
Intended Use	Educational and Research Application	Research	Clinical Application

consistent waveform outputs and reliable BPM estimates generated from the digital stethoscope during the experimental tests. Additionally, when all the preprocessing techniques (e.g., filtering, denoising and segmentation) were applied to the heartbeat signals before they were analyzed and classified according to the presence of heart murmurs, the quality of the waveform significantly improved, and an overall reduction in the amount of noise present in the output of the analysis occurred.^[6,7] The development of machine learning-based murmur classification techniques has also substantially enhanced the ability of digital stethoscope systems to provide preliminary automated classifications of heart sounds based on a phonocardiogram, as described in some of the current literature related to the intelligent analysis of heart sounds.^[8,9] Other features of the digital stethoscope system, including an integrated dashboard for displaying patient data and monitoring capabilities, integrated patient record management capabilities, CSV export functions, and automated generation of PDF reports, all contribute to making the digital stethoscope a useful biomedical education tool, a prototyping system for further development, and an intelligent medical monitoring system.^[11]

Despite showing hope through experimental outcomes, limitations of the developed system exist when the system is tested. The combination of a low-cost electret microphone and a KY-037 sound sensor resulted in lower sensitivity to clinical-grade digital stethoscopes when the detection of low-intensity heart murmurs and abnormal heart sounds was detected.^[12,14] Additionally, the acquired heart sounds did not show quality during actual monitoring because of environmental noise, improper placement of the chest piece, and improper handling of the system by the user. Prior to clinical validation, the intended use of this system will be to

educate, research, and complete some preliminary screening rather than direct medical diagnosis. Future improvements may involve advanced denoising algorithms, heart sound classification based on deep learning techniques, healthcare devices that are/or may become wearable, and IoT-enabled telemedicine systems, which will allow for remote patient monitoring and other forms of intelligent healthcare.^[2,8]

3.5 Future scope

The inclusion of IoT technologies and cloud-based physicians into the digital stethoscope system will enhance remote patient monitoring and telemedicine services.^[2,13] The future enhancements may include wireless communication support, mobile application integration, cloud database connectivity, and wearable embedded healthcare devices in effort to allow for continuous real-time cardiac monitoring. Furthermore, advanced artificial intelligence methods will include CNNs, hybrid deep learning models, and intelligent phonocardiogram analysis systems, which can further enhance the accuracy and robustness of abnormal heart sound classification and automated cardiac diagnosis.^[3,8,9]

4. Conclusion

The paper describes an affordable and simple system designed for heart sound acquisition and analysis using an Arduino-based intelligent digital stethoscope. The system uses a modified KY-037 sound sensor and an external electret microphone to acquire heart sounds in real-time, produce waveforms of heart activity in real-time, allow the estimation of heartbeats per minute (BPM) via a machine learning classification system based on the features of the sound signal, and are visualized using the Streamlit web application. The combination of

filtering techniques and adaptive peak detection, random forest classification and Streamlit-based visualization created an intelligent system that would provide efficient and effective methods to monitor and process cardiac sounds in real-time. The system can be used in both educational and research settings for both cardiac monitoring applications due to its low-cost and portable nature. The addition of a database for long-term storage of heart sounds, CSV export functionality, and automated generation of PDF documentation improves the intelligent monitoring ability of the system. Although the completion of the system will not be able to directly diagnose patients clinically, it demonstrates the utility of low-cost, embedded biomedical systems for preliminary cardiac monitoring and the research of healthcare technology.

CRedit Author Contribution Statement

Nafisa Mapari: Formal analysis, Methodology, Project administration. **Alefiya Managori:** Conceptualization, Formal analysis, Data curation, Software, Visualization, Writing-Original draft preparation, Writing- Review & editing. **Hajra Naik:** Data curation, Methodology, Investigation. **Tasmiya Kandhal:** Validation, Writing-Review & editing. **Monika Chaurasiya:** Methodology, Resources, Writing-Review & editing. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement

The dataset used in this study is publicly available through the [PhysioNet CirCor DigiScope Phonocardiogram Dataset repository](#), (Ref. 11). The developed source code, preprocessing methods, and implementation details are available from the corresponding author upon reasonable request.

Conflict of Interest

The authors received no specific financial support for the research, authorship, or publication of this work.

Funding Declaration

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Artificial Intelligence (AI) Use Disclosure

The authors declare that artificial intelligence (AI)-assisted tools were used only for language refinement, grammar improvement, and manuscript structuring purposes during the preparation of this work. All technical content, experimental implementation, results, and interpretations were independently developed and verified by the authors.

References

- [1] J. McGee, Evidence-Based Physical Diagnosis, 4th edition. Philadelphia, PA, USA: Elsevier, 2018.
- [2] S. Abdulmalek, A. Nasir, W. A. Jabbar, M. A. M. Almuhaaya, A. K. Bairagi, M. A.M. Khan, S. H. Kee, IoT-based healthcare-monitoring system towards improving quality of life: A review, *Healthcare*, 2022, **10**, 1993, doi: 10.3390/healthcare10101993.
- [3] A. Chorba, J. Shapiro, A. LeCun, Y. Bengio, Deep learning algorithm for automated cardiac murmur detection via a digital stethoscope platform, *Journal of the American Heart Association*, 2021, **10**, e019905. doi: 10.1161/JAHA.120.019905.
- [4] H. Liang, S. Lukkarinen, I. Hartimo, Heart sound segmentation algorithm based on heart sound envelopogram, *Computers in Cardiology*, 1997, **24**, 105–108, doi: 10.1109/CIC.1997.647841.
- [5] D. Castaneda, A. Esparza, M. Ghamari, C. Soltanpur, H. Nazeran, A review on wearable photoplethysmography sensors and their potential future applications in health care, *International Journal of Biosensors & Bioelectronics*, 2018, **4**, 195–202, doi: 10.15406/ijbsbe.2018.04.00125.
- [6] D. Springer, L. Tarassenko, G. Clifford, Logistic regression-HSMM-based heart sound segmentation, *IEEE Transactions on Biomedical Engineering*, 2016, **63**, 822–832, doi: 10.1109/TBME.2015.2475278.
- [7] C. Liu, D. Springer, Q. Li, B. Moody, R. A. Juan, F. J. Chorro, F. Castells, J. M. Roig, I. Silva, A. E. Johnson, Z. Syed, An Open access database for the evaluation of heart sound algorithms, *Physiological Measurement*, 2016, **37**, 2181–2213, doi: 10.1088/1361-6579/37/12/2181.
- [8] K. N. Khan, F. A. Khan, A. Abid, T. Olmez, Z. Dokur, A. Khandakar, M. E. Chowdhury, M. S. Khan, Deep Learning based classification of unsegmented phonocardiogram spectrograms leveraging transfer learning, *Physiological measurement*, 2021, **42**, 095003, doi: 10.1088/1361-6579/ac1d59.
- [9] M. Kalimuthu, C. Hemanth, A comparative analysis of machine learning and deep learning approaches for phonocardiogram classification using dataset integration, *IEEE Access*, 2025, **13**, 170619–170635, doi: 10.1109/ACCESS.2025.3612392.
- [10] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Moody, C. K. Peng, H. E. Stanley, PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals, *Circulation*, 2000, **101**, e215–e220, doi: 10.1161/01.CIR.101.23.e215.
- [11] J. Oliveira, F. Ferreira, C. Pereira, A. Silva, R. Costa, M. Fonseca, A. Santos, H. Gamboa, The CirCor DigiScope Phonocardiogram Dataset, In IEEE Conference, PhysioNet, 2022, doi: 10.13026/tshs-mw03.
- [12] M. A Reyna, Y. Kiarashi, A. Elola, J. Oliveira, F. Renna, A. Gu, E. A. Perez Alday, N. Sadr, A. Sharma, J. Kpodonu, S. Mattos, Heart murmur detection from phonocardiogram

recordings: The George B. Moody PhysioNet Challenge 2022, *Computing in Cardiology*, 2022, **49**, 1–4, doi: 10.1371/journal.pdig.0000324.

[13] M. Kalimuthu, C. Hemanth, Preliminary Study on real-time phonocardiogram signal acquisition and analysis using machine learning and IoMT for digital stethoscope, *IEEE Access*, 2025, **13**, 68682–68709. doi: 10.1109/ACCESS.2025.3560763.

[14] Arduino, Arduino UNO Rev3 Technical Specifications, Arduino Documentation, 2023. Accessed: May 20, 2026,

available at: <https://docs.arduino.cc/hardware/uno-rev3/>

[15] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, O. Nieto, Librosa: Audio and music signal analysis in python, *SciPy*, 2015, **7**, 18–25. doi: 10.25080/Majora-7b98e3ed-003.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: Machine learning in python, *Journal of Machine Learning Research*, 2011, **12**, 2825–2830.

Publisher Note: The views, statements, and data in all publications solely belong to the authors and contributors. GR Scholastic is not responsible for any injury resulting from the ideas, methods, or products mentioned. GR Scholastic remains neutral regarding jurisdictional claims in published maps and institutional affiliations.

Open Access

This article is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License, which permits the non-commercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as appropriate credit to the original author(s) and the source is given by providing a link to the Creative Commons License and changes need to be indicated if there are any. The images or other third-party material in this article are included in the article's Creative Commons License, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons License and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this License, visit: <https://creativecommons.org/licenses/by-nc/4.0/>

© The Author(s) 2026