UNIVERSITY
OF TURKU

The Impact of ECG Compression on Deep Learning-Based Classification

Health Technology

Master's Degree Programme in Health Technology

Department of Computing, Faculty of Technology

Master of Science in Technology Thesis

Author:

MD Monzurul Islam Rabu

Supervisors:

Assistant Professor Matti Kaisti, PhD

Senior Researcher Jiawei Yang, PhD

June 2025

Master of Science in Technology Thesis

Department of Computing, Faculty of Technology

University of Turku

Subject: Health Technology

Programme: Master's Degree Programme in Health Technology
Author: MD Monzurul Islam Rabu
Title: The Impact of ECG Compression on Deep Learning-Based Classification

Number of pages: 65 pages

Date: June 2025

Abstract.

Edge-based ECG devices and real-time continuous ECG monitoring can play an important role in reducing cardiovascular disease (CVD)-related deaths. With recent advancements in artificial intelligence (AI), there is a growing need to deploy arrhythmia classification models on edge devices, especially in resource-constrained environments such as wearable monitors and home-based ECG patches.

However, ECG signals are high-frequency time-series data, and they generate large volumes of information even in short recordings. Continuous monitoring can easily result in hundreds of megabytes to gigabytes of data per day, making it difficult to store, transmit, or process such data on low-power or low-bandwidth systems. Deep learning models require substantial computational resources when processing high-dimensional raw data, which many edge devices, such as microcontrollers and ARM Cortex-M CPUs, cannot support efficiently. As a result, inference times increase, and the system may fail to deliver timely alerts or classification results. To address these limitations, adopting ECG data compression techniques is crucial. However, ECG signal compression always carries the risk of losing substantial, clinically significant data. Therefore, after compression, the machine learning-based classification models might fail to classify accurately. It is important to determine the exact impact of ECG signal compression on machine learning-based classification models. In this thesis, we investigated the effect of different ECG compressions on machine learning-based ECG arrhythmia classification models.

Keywords: ECG Signal Processing, Arrhythmia Classification, Edge Computing, ECG Compression, Machine Learning, Deep Learning, Wearable Devices.

Table of Contents

1 Introduction

1.1 Background and Motivation

Cardiac arrhythmias are among the leading causes of cardiovascular-related morbidity and mortality worldwide. Recent progress in analyzing ECG signals has significantly improved the detection and classification of cardiac arrhythmias. Specifically, the deep learning approaches can achieve cardiologist-level performance in identifying various rhythm abnormalities from single-lead ECGs [1].

Accurate diagnosis and continuous monitoring of cardiac arrhythmias through electrocardiograms (ECGs) is critically important. ECG data can be considered moderately resource-intensive depending on the application, particularly when dealing with long-term monitoring, high-resolution signals, or large-scale AI training datasets. The widespread adoption of wearable devices and telemedicine systems has significantly increased the volume of ECG data generated, creating an urgent need for efficient compression techniques. Such techniques are essential not only for managing data storage and transmission but also for enabling sophisticated AI-based arrhythmia detection models to operate effectively in resource-constrained environments. However, compression risks the loss of clinically vital ECG features, such as QRS morphology, ST segments, and P-waves, potentially impacting diagnostic accuracy [2].

A variety of compression techniques that have been proposed primarily can be categorized into four methodological groups: (1) Lossless Methods, (2) Deep Learning-Based Methods, (3) Transform-Based Methods, and (4) Hybrid Compression Methods. Although previous studies have demonstrated promising results in terms of compression ratio (CR), Percent Root Mean Square Difference (PRD), and Signal-to-Noise Ratio (SNR), there remains uncertainty regarding how these ECG compression methods affect the diagnostic accuracy and computational efficiency of modern arrhythmia classifiers. The validation of compression methods from the perspective of usability in the machine learning-based classification models remains an active area of research [3]. Many of the existing compression techniques were developed before the widespread adoption of machine learning based ECG arrhythmias classification. Therefore, during the development of these compression algorithms, their compression effects on Machine Learning performance were not thoroughly evaluated.

This thesis addresses this gap by systematically evaluating how ECG compression methods influence the diagnostic accuracy and computational efficiency of two representative machine-learning-based classifiers: a hybrid CNN-LSTM deep learning model and a traditional RF-SVM ensemble classifier.

Understanding these impacts is crucial in deploying machine learning based models in resource-constrained environments, where optimal trade-offs between data fidelity, diagnostic accuracy, and computational efficiency directly influence clinical outcomes. The findings of this research will contribute to optimizing ECG data management, enhancing real-time clinical decision-making and patient monitoring, improving computational efficiency, ensuring diagnostic reliability, and providing clear design guidelines for deploying effective AI-based ECG monitoring systems on edge devices.

1.2 Problem Statement

While extensive research has developed numerous ECG compression methods optimized for traditional signal fidelity metrics—such as Compression Ratio (CR), Percent Root Mean Square Difference (PRD), Signal to Noise Ratio (SNR) these metrics alone cannot effectively determine how compression techniques preserve clinically significant ECG features, particularly when the compressed data is used by machine learning-based arrhythmia classifiers. Again, these compression techniques have not been extensively evaluated on edge-based devices. Therefore, there are very limited clues available about how these compressions impact the classification models in terms of accuracy and inference time, especially in edge deployments. Aggressive compression on the ECG signal may cause substantial loss of clinically valuable information on the signal. Consequently, when this compressed and then decompressed signal is used in the classification model, this compression impact might degrade the diagnostic accuracy of machine learning based automated classifiers, increasing the risk of misdiagnosis. On the other hand, inadequate or non-optimized compression of ECG data may lead to increased inference latency and power consumption due to additional computational overhead, making the model less suitable for deployment in resource-constrained wearable devices and telemedicine systems [4].

To our knowledge, no prior work has systematically studies thoroughly investigated how different ECG compression techniques impact the performance of contemporary machine learning based arrhythmia detection models when compressed data is used either as test data or training data, or in both test and training data. To tackle this critical gap, this thesis seeks to address the following research question through carefully designed experimental scenarios.:

How do different ECG compression methods influence diagnostic accuracy and inference time of machine learning-based ECG arrhythmia classifiers, particularly in computationally constrained environments?

1.3 Objective of the Study

The primary objective of this research is to systematically evaluate how ECG compression techniques influence the accuracy and computational efficiency of modern arrhythmia classifiers. Specific objectives aligned with experimental scenarios include:

1. Conduct a comprehensive literature review to classify recent ECG compression techniques into the four categorized methodological groups.
2. Identify the top-performing technique within each methodological group by comparing standard compression evaluation metrics recorded in the recent corresponding publication.
3. Develop and benchmark two representative arrhythmia classification models—one traditional (RF-SVM) ensemble model and one deep learning-based (CNN-LSTM) model—using the MIT-BIH arrhythmia dataset.
4. Implement selected compression methods and apply them to generate compressed versions of the MIT-BIH ECG dataset.
5. Evaluate the performance impact of applying each compression technique on both classification models across three distinct experimental scenarios for each compression technique (original-original, original-compressed, compressed-compressed ECG dataset for the training and test of the models).

6. To provide practical guidance for the design of compression-aware and AI-compatible ECG monitoring systems, particularly suited for wearable devices and telemedicine platforms operating in resource-constrained environments.

In this study, we distinctly evaluated different ECG compression techniques specifically from the perspective of their impacts on modern ML-based arrhythmia classifiers. We chose CNN-LSTM and RF-SVM ensemble models because they represent robust, widely validated approaches in ECG classification. This thesis examined how ECG signal compression influences classification accuracy across deep learning based and traditional machine learning based ECG arrhythmia classification model. While both deep learning method (CNN-LSTM) and traditional machine learning method (RF-SVM ensemble model) are explored, our study primarily emphasized the impact of ECG signal compression on deep learning-based classification due to its emerging importance and superior potential in clinical applications.

2 Literature Review

2.1 Electrophysiology and Clinical Relevance of ECG Signals

2.1.1 Physiological Basis of ECG Signals

The heart's intrinsic conduction system consists of specialized pacemaker cells and conducting cells. This architecture enables autonomous rhythmic contraction independent of neural stimulation. This system generates coordinated electrical impulses that propagate sequentially through the cardiac chambers, initiating synchronized atrial and ventricular contractions [5]. Surface electrocardiography (ECG) captures these electrical patterns noninvasively, serving as a critical diagnostic tool for assessing cardiac electrophysiology and detecting arrhythmia.

Electrical activity of the heart originates in a small cluster of modified cardiac muscle cells known as pacemaker cells. The SA node initiates each heartbeat by generating electrical impulses at a rate of 60 to 100 times per minute in healthy individuals. These impulses spread across the atria, triggering their depolarization and contraction. The signal then reaches the atrioventricular (AV) node, situated near the AV valve. The atrioventricular (AV) node briefly slows the electrical signal, ensuring the atria complete their contraction before ventricular depolarization begins [6]. The impulse then moves through the bundle of His, splitting into the right and left bundle branches, which guide the electrical wave toward the heart's apex. The impulse is finally distributed throughout the ventricular myocardium via Purkinje fibers, resulting in synchronized ventricular contraction [7].

Pacemaker cells in the SA node lack a true resting potential; instead, spontaneous depolarization occurs due to "funny" sodium currents until voltage-gated calcium channels open at threshold, causing rapid depolarization [8]. Repolarization follows via potassium efflux, and ion pumps reset the gradients. In contrast, contractile myocytes have a stable −90 mV resting potential and are activated by depolarization from adjacent cells. Fast Na^+ influx initiates depolarization, followed by a plateau phase from L-type Ca^{2+} influx, which triggers calcium-induced calcium release for contraction, before repolarization via K^+ efflux [9]. The coordinated depolarization and repolarization of numerous cardiac cells produce small electrical signals within the heart. These signals travel through cardiac tissues and surrounding body fluids, reaching the skin surface. Electrodes placed on the body detect these tiny electrical changes, which are then amplified and displayed as standard ECG waveforms [10].

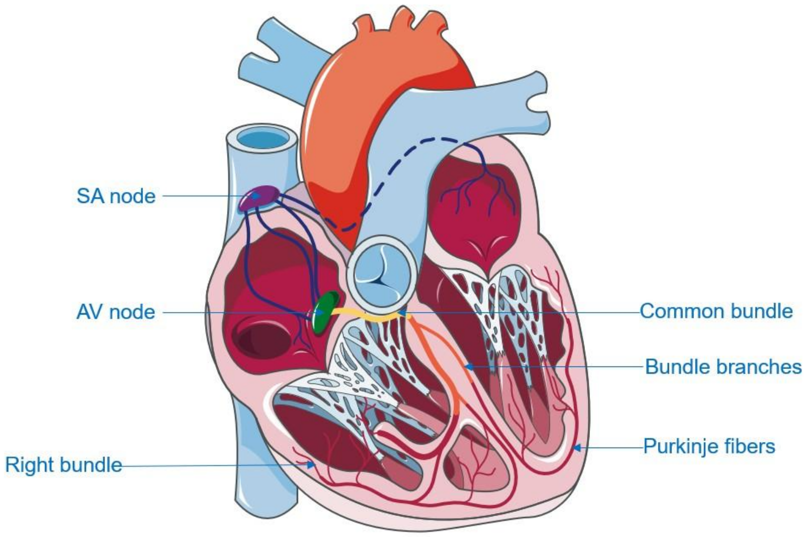

Figure 1 Diagram of the cardiac conduction system, reconstructed from Servier Medical Art, licensed under CC BY 4.0. [11]

2.1.2 ECG Signal Acquisition and Lead Configuration

The 12-lead ECG system, considered the clinical benchmark for recording the heart's electrical activity, uses 10 electrodes placed strategically on the limbs and chest; six of these leads (I, II, III, aVR, aVL, and aVF) specifically capture limb-based views [12]. An imaginary triangle can be formed with leads I, II, and III; this is theoretically called Einthoven's triangle. The other six leads which are positioned in the chest, these leads (V1 to V6) give a view across the heart from side to side. While this configuration enables comprehensive cardiac evaluation, it also introduces significant signal heterogeneity—a critical consideration for ECG compression.

For instance, limb leads—particularly leads II and aVF—tend to capture low-amplitude P waves, which are indispensable for detecting atrial arrhythmias. In contrast, precordial leads, especially V1 through V5, record high-frequency QRS complexes that are vital for diagnosing ventricular abnormalities[12]. Noisier and lower-amplitude limb leads may benefit from bit-depth allocation, whereas precordial leads require high-frequency preservation to maintain the integrity of QRS complexes. Nowadays, modern edge-based devices like wearable or patch ECGs use fewer electrodes—sometimes even just one. Instead of showing 12 full leads like a standard ECG, these devices record limited electrical views of the heart [13].

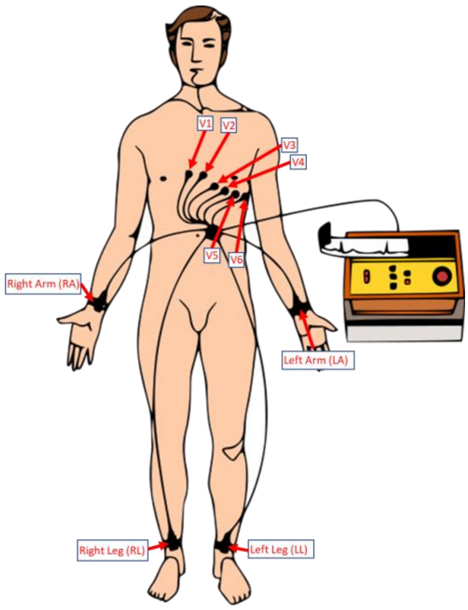

Figure 2 Standard 12-lead ECG electrode placement. Adapted from Tazie25, Wikimedia Commons, licensed under CC BY-SA 4.0.

2.1.3 Clinical Components of the ECG Signal

Each component of the ECG waveform reflects a specific electrical event in the cardiac cycle and provides valuable diagnostic insight. The P wave depicts atrial depolarization initiated by the sinoatrial (SA) node, typically beginning approximately 40 milliseconds after SA node activation. Under normal conditions, the P wave appears upright in leads I and II, inverted in lead aVR, and is smooth, monophasic, and less than 120 milliseconds in duration [12]. Deviations in P wave morphology may indicate atrial abnormalities. For example, right atrial enlargement often results in tall, peaked P waves, while left atrial enlargement is associated with prolonged, notched, or bifid P waves [14].

From the QRS complex we can get an insight of the rapid electrical depolarization occurring within the ventricles. It begins at the atrioventricular (AV) node and continue through the His–Purkinje system and ventricular myocardium. The QRS complex consists of three parts: the Q wave, an initial negative deflection representing septal depolarization; the R wave, a subsequent positive deflection indicating depolarization of the main ventricular mass; and the S wave, a negative deflection following the R wave, corresponding to late ventricular depolarization at the base [14]. The morphology of the QRS complex varies across precordial leads. A normal QRS duration is less than 120 milliseconds; prolongation may suggest intraventricular conduction delays, such as left or right bundle branch block [15].

The ST segment begins immediately after the QRS complex and continues until the start of the T wave. It reflects the plateau phase of the ventricular action potential and corresponds to the period of early ventricular repolarization when little to no net electrical activity is detected. A normal ST segment is typically flat or isoelectric. Deviations from this baseline, such as ST elevation or depression, are critical indicators of acute myocardial infarction or myocardial ischemia, respectively [16].

The T wave illustrates ventricular repolarization. Although repolarization occurs in the opposite direction of depolarization, the T wave is usually upright in most ECG leads due to the unique vector of electrical recovery. A normal T wave should be asymmetric, smooth, and rounded. Abnormalities in the T wave may reveal various conditions. The tall, peaked T waves are often associated with hyperkalemia, while inverted T waves may reflect ischemia, previous infarction, or ventricular strain [17].

In addition to the primary waveform components, specific intervals and segments offer further clinical information. The P–R interval, measured from the onset of the P wave to the beginning of the QRS complex, represents atrial depolarization and conduction through the AV node. The Q–T interval, which spans from the beginning of the Q wave to the end of the T wave, reflects total ventricular activity, encompassing both depolarization and repolarization. The duration of Q–T interval varies with heart rate. The R–R interval, measured between successive R waves, represents one full cardiac cycle and is commonly used to assess heart rate and rhythm regularity. Finally, the ST segment is particularly sensitive to ischemic changes and remains a key area of focus in diagnosing acute coronary syndromes. Understanding these components and their clinical significance provides a basis for accurate ECG interpretation and diagnosis [18].

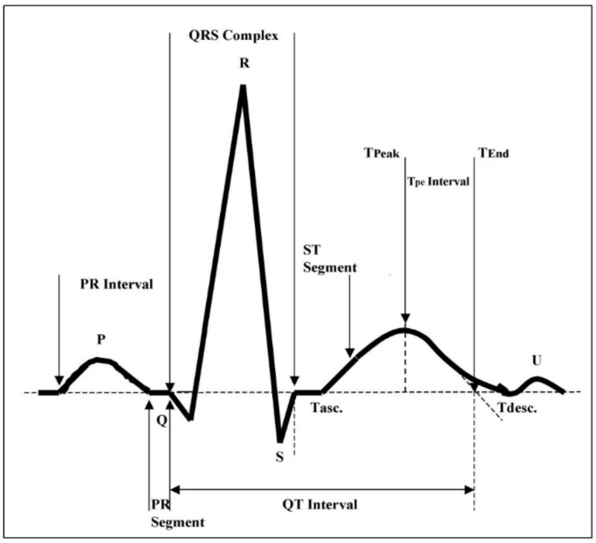

Figure 3 ECG waveform showing a cardiac cycle. Reused from Pater (2005) [19]

2.1.4 Cardiac Arrhythmias and Their ECG Signatures

Cardiac arrhythmia occurs when the heart's electrical signals are disrupted. These disturbances can start in any part of the heart and are identified based on their origin and ECG characteristics. Recognizing these patterns accurately is essential for proper diagnosis and treatment [20].

Cardiac arrhythmias broadly classify into three pathophysiological categories: Disorders of impulse formation involve abnormal initiation rates or sites of electrical impulses, conduction abnormalities result from disrupted or delayed electrical pathways, and triggered activity involves premature impulses due to after-depolarizations. Examples of common arrhythmias include sinus arrhythmia, and sinus bradycardia (<60 bpm).

Sinus tachycardia, defined as a heart rate above 100 bpm, usually results from increased sympathetic activity. Early (ectopic) heartbeats can originate from the atria (PACs), AV junction (PJCs), or ventricles (PVCs). PACs present with early upright P waves and narrow QRS complexes; PJCs typically show narrow QRS complexes with absent or inverted P waves; PVCs display wide QRS complexes without preceding P waves and often feature opposite-directed T waves. [21].

Slow arrhythmias (bradyarrhythmias) include sinus bradycardia and advanced AV blocks. Fast arrhythmias (tachyarrhythmias), like SVTs, originate above the ventricles, generally exceed 150 bpm, and produce narrow QRS complexes with difficult-to-see or merged P waves. AVNRT and AVRT are common types. ECG patterns of these arrhythmia reveal their underlying mechanisms: enhanced automaticity, reentry, or triggered activity.

2.1.5 Importance of Morphological Features in ECG Data Compression

Recognition of ECG waveform signatures is essential for timely and accurate intervention, whether through manual interpretation or automated classification. When compressed ECG data is used in automated arrhythmia detection systems, particularly with lossy compression techniques, there is a risk of degrading clinically relevant features. Arrhythmia classifiers often rely on precise P wave morphology and QRS complex fidelity; compression artifacts that alter these elements may significantly reduce diagnostic accuracy[22].

Diagnostic performance depends on preserving three key aspects: (1) temporal correlations, such as rate-corrected QT intervals; (2) spatial patterns, such as precordial R-wave progression; and (3) amplitude ratios between waveform components. Compression-induced disruptions—such as asynchronous distortion of PR and ST segments—can compound diagnostic errors beyond isolated feature degradation [22].

Feature	**Clinical Significance**	**Compression Risks**
P Wave	Atrial depolarization. Absence indicates atrial fibrillation. Morphology changes suggest atrial enlargement.	Amplitude attenuation (>20%) may obscure atrial activity. Time misalignment ($\pm$5 ms) can distort P-wave onset.
QRS Complex	Ventricular depolarization. Width >120 ms suggests conduction delays. Morphology used to diagnose bundle branch blocks.	Temporal quantization can artificially widen QRS duration. High-frequency loss may blunt R-wave peaks.
ST Segment	Ventricular plateau phase. Elevation or depression indicates ischemia. Slope changes observed in pericarditis.	Smoothing may mask ischemic deviations ($\pm$100 μV). Baseline correction may distort ST level.
T Wave	Ventricular repolarization. Inversion may indicate ischemia. Peaking suggests hyperkalemia.	Amplitude reduction may obscure electrolyte abnormalities. Phase distortion can introduce artificial notching.

Figure 4 ECG waveform components: clinical significance and compression-related risks.

2.2 Theoretical Background of ECG Compression

2.2.1 Introduction to ECG Compression

ECG compression transforms an original discrete-time signal x[n] of length N_o into a compressed representation c[k] of length Nc where Nc $\ll N_o$. Compression efficiency is quantified by the compression ratio. Both lossy and lossless compression techniques can be employed to minimize storage and bandwidth requirements of ECG signal while preserving diagnostically significant features [23]. The theoretical basis of ECG signal compression derives from information theory, encompassing key principles such as entropy coding, transform-based redundancy reduction, and rate-distortion trade-offs.

2.2.2 Digital Representation and Redundancy in ECG Signals

The digital representation of ECG signals begins with analog-to-digital conversion (ADC), where continuous waveforms are discretized into numerical samples at a fixed sampling rate (typically 250-1000 Hz in clinical settings). Each sample is quantized using a uniform bit-depth (e.g., 8, 12, or 16 bits per sample), resulting in a fixed-length pulse-code modulation (PCM) format. While PCM is computationally simple, it is inherently inefficient for ECG signals because it fails to exploit temporal redundancies (e.g., quasi-periodic R-peaks) and amplitude correlations (e.g., predictable P-wave and T-wave morphologies) [24].

This issue is intensified by ECG signal non-stationarity, where features like heart rate and noise levels vary over time, making static compression strategies suboptimal. Advanced compression methods address these limitations through adaptive quantization (allocating more bits to clinically significant regions), entropy coding (exploiting statistical redundancies), and context-aware strategies (adjusting compression parameters dynamically). These approaches significantly reduce bitrate while maintaining diagnostic fidelity, making them highly suitable for telemedicine and wearable healthcare applications. This thesis comprehensively reviews recent ECG compression techniques, identifies the most promising methods from the literature.

2.2.3 Entropy and Compression Limits

Entropy-based approaches offer superior compression efficiency for ECG signals than PCM. Entropy, defined by Shannon's source coding theorem, establishes the theoretical minimum for average bit length per symbol in lossless compression as:

$$H(X) = -\sum p_i \log_2 p_i \tag{2.1}$$

pi is the probability of symbol occurrence. Achieving a bit rate near this entropy value is the fundamental goal of efficient ECG compression [25].

To approach this theoretical limit practically, variable-length coding (VLC) techniques dynamically assign shorter codewords to high-probability symbols and longer ones to low-probability symbols, minimizing the expected code length:

$$L = \sum_{i=1}^{n} p_i \; l_i \tag{2.2}$$

pi is the probability of symbol, and li is its codeword length.

VLC algorithms—such as Huffman coding (for fixed symbol probabilities), arithmetic coding (for adaptive scenarios), and asymmetric numeral systems (ANS, state-driven compression)—are especially effective when combined with preprocessing steps like linear predictive coding (LPC) or first-order differencing. These preprocessing methods transform ECG residual signals toward distributions (e.g., Laplacian) favorable for entropy coding. Such entropy-driven approaches are particularly crucial for preserving diagnostically significant low-amplitude features such as P-waves (~0.1–0.3 mV).

2.2.4 Rate-Distortion Theory

In Lossy ECG compression Rate-Distortion theory provides a framework to balance bitrate and signal fidelity. The trade-off is expressed through mutual information:

$$\mathrm{I(X;Y) = H(X) - H(X \mid Y)} \tag{2.3}$$

here X is the original signal, and Y denotes its reconstructed counterpart. For Gaussian-distributed signals, such as certain ECG segments, the theoretical limit for achievable compression with a specified distortion D can be calculated from rate-distortion function:

$$R(D) = \frac{1}{2}\log\left(\frac{\sigma^2}{D}\right) \tag{2.4}$$

where $\sigma 2$ is the variance of the signal, and D represents the average distortion allowed. This function specifies the minimum number of bits per sample needed to encode a Gaussian signal at a given level of distortion. Transform coding techniques such as Discrete Cosine Transform (DCT) operationalize R-D theory by concentrating signal energy into fewer coefficients. Consequently, this enables efficient bit allocation prioritizing clinically significant ECG features such as the QRS complex or the ST segment [24]. In practical ECG applications, distortion metrics derived from R-D analysis must carefully incorporate clinical interpretability and compatibility with downstream automated diagnostic algorithms.

2.3 Evaluation Metrics for Compression Performance

2.3.1 Compression Ratio (CR)

Compression Ratio (CR) quantifies the effectiveness of data reduction achieved by a compression algorithm. The ratio between the number of bits in the original signal and the number of bits in the compressed signal is defined as the compression ratio:

$$\mathrm{CR} = \frac{\mathrm{N_{original}}}{\mathrm{N_{compressed}}} \tag{2.5}$$

here N original is the number of bits in the original signal, and N compressed is the number of bits in the compressed signal [26]. Higher CR values indicate more efficient compression. However, extremely high CR can compromise signal quality, particularly in lossy compression schemes.

2.3.2 Percent Root Mean Square Difference (PRD)

Distortion between the original and reconstructed signals is measured using the Percent Root Mean Square Difference (PRD):

$$\text{PRD} = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \widehat{x_i})^2}{\sum_{i=1}^{N} x_i^2}} \times 100\% \tag{2.6}$$

Here, x_i is the original signal, $\widehat{x_i}$ is the reconstructed signal after decompression, and N is the total number of samples [27]. A lower PRD value implies better signal reconstruction fidelity. PRD is especially critical in ECG analysis, where even small distortions in P, QRS, or T waves may affect diagnostic interpretation [28].

2.3.3 Signal-to-Noise Ratio (SNR)

The Signal-to-Noise Ratio (SNR) evaluates the strength of the original signal relative to the distortion (error) introduced by compression:

$$\text{SNR (dB)} = 10 \cdot \log_{10}\left(\frac{\sum_{i=1}^{N} x_i^2}{\sum_{i=1}^{N}(x_i - \widehat{x_i})^2}\right) \tag{2.7}$$

Higher SNR values indicate that the reconstructed signal closely resembles the original, with minimal noise or distortion. SNR is commonly used alongside PRD to assess waveform preservation, particularly for subtle features like ST-segment deviations or P-wave morphology [29].

2.3.4 Application-Dependent Considerations

While the above metrics provide standardized measures, the optimal balance between compression and fidelity depends on the target application. For instance, wearable devices and mobile ECG monitors often prioritize high CR to reduce data transmission costs, even if mild signal distortion occurs. In contrast, diagnostic systems in clinical environments require high-fidelity reconstructions (low PRD and high SNR) to preserve waveform morphology for accurate interpretation [30].

2.3.5 ECG Compression Paradigms & Trade-offs

Lossless ECG compression techniques (e.g., entropy coding) preserve diagnostic fidelity but achieve limited compression ratios (typically 2:1–4:1), while lossy techniques (e.g., transform coding) offer higher compression (5:1–30:1) but require careful distortion control.

Ensuring decodability is essential in ECG compression systems. Variable-length codes (VLCs) must satisfy the prefix-free property—no codeword is a prefix of another—as guaranteed by the Kraft-McMillan inequality

$$\sum_{i=1}^{n} 2^{-l_i} \leq 1 \tag{2.8}$$

This ensures unambiguous decoding, which is critical in real-time ECG telemetry and embedded systems such as Holter monitors or wearable patches.

Implementations must also meet real-time constraints such as low latency (≤10 ms per window), minimal memory footprint (≤8 KB RAM), and energy efficiency (≤1 µJ/sample). Efficient decoding architectures often use finite-state machines (FSMs) for Huffman/ANS decoding or parallelized arithmetic coding, as demonstrated in FDA-cleared devices like the Zio® patch [31].

2.4 Core Compression Techniques

To provide a structured and comprehensive overview of existing ECG compression methods, we categorize ECG signal compression into four broad categories. Although this classification is not universally standardized, it offers a practical framework for evaluating and comparing contemporary compression techniques. We tried to incorporate most of the ECG Compression techniques into these four categories Lossless Compression Methods, Deep Learning-Based Methods, Transform-Based Methods and Hybrid Compression Methods. In the hybrid Compression Techniques, we put all the compression methods that combine two or more techniques as wavelet transforms paired with entropy coding or neural networks integrated with traditional approaches. It is true that in this study certain techniques may conceptually overlap among categories, however this classification provides a logical and practical framework for systematically analysing diverse ECG compression methodologies.

2.4.1 Lossless Compression Techniques

Overview & Theoretical Context

Entropy coding and lossless compression methods such as Arithmetic coding are essential to preserve the critical diagnostic details of ECG signals. Due to their exact signal fidelity, these methods are preferred in clinical settings, such as patient monitoring or archival storage. Recent advancements, including the computationally efficient Asymmetric Numeral Systems (ANS), have demonstrated suitability for embedded applications, addressing practical limitations like computational resources and battery life constraints in medical devices [32].

Techniques and Taxonomy

There are a variety of entropy and lossless coding techniques for ECG signal compression. Each technique offers different advantages in terms of algorithmic complexity, compression ratio, and suitability for hardware implementation. Huffman coding is one of the most classic and widely used approaches. Minimum variance of Huffman coding shown further improves compression efficiency by optimizing the code lengths based on symbol probability distributions. Arithmetic coding techniques are another powerful alternative. This technique encodes entire sequences as intervals on the number line rather than symbol by symbol. More recently, Asymmetric Numeral Systems (ANS) have emerged as a modern entropy coding strategy that combines the compression efficiency of arithmetic coding with the computational simplicity of table-based methods. Due to their speed and low memory requirements, ANS algorithms are particularly well-suited for embedded ECG applications. Golomb and Golomb-Rice coding techniques, often applied in conjunction with predictive models, are efficient for signals exhibiting geometric-like distributions and are frequently used in combination with run-length encoding (RLE) to compress repetitive waveform patterns. RLE when applied to flat or low-variance segments of ECG signals and integrated with Golomb coding the combination become particularly efficient in capturing repetitive structures [33].

Best Performing Lossless Based Techniques in Literature

Among the reviewed recent publications on lossless ECG signal compression, we observed that an effective lossless compression can achieve the highest compression ratios up to 6.52:1 on PTBDB and 3.82:1 on MITDB—while maintaining zero distortion. Some of the lossless compression techniques have relative limitations in embedded applicability. We observed comparatively one of the highest compression ratios is achieved through the lossless compression technique proposed by Banerjee and Singh [35]. In this paper, the author used a lossless compression method combining hierarchical difference coding and sequential repetition optimization, with buffer arrays to reduce intrabeat redundancy. The method achieved high compression ratios, reaching up to 6.52 for PTBDB ECG, 3.82 for MITDB ECG, and 2.49 for MIMIC-II PPG signals, all without any loss of signal fidelity. The following Table 1 summarizes some of the effective lossless compression techniques and their strengths, along with the performance metrics.

Table 1 Comparison of Lossless ECG Compression Techniques with Reported CR and Strengths

Technique	CR / PRD Range	Key Strength(s)	Reference
Adaptive Linear Prediction (intra-channel) + Multi-channel Linear Prediction (inter-channel) + Adaptive Golomb-Rice Coding	2.89 (single channel) 4.07 (multi-channel) MIT-BIH, PTB	Real-time embedded implementation	Tsung-Han Tsai & Fong-Lin Tsai (2020) [34]
Second Order Delta Encoding + Run Length Encoding + Buffer array	CR 3.82 (MITDB) CR 6.52 (PTBDB) CR 2.49 (MIMIC-II)	Independent of sampling frequency and resolution	Banerjee, S., & Singh, G. K., 2023 [35]
Error Based Adaptive Linear Prediction + Modified Golomb–Rice	3.52 (MIT-BIH) 3.75 (EDB) 3.16 (PTBDB)	Absolutely lossless & Real-time capable	Bannajak, K., Theera-Umpon, N., & Auephanwiriyakul, S. (2023) [36]

2.4.2 Deep Learning-Based Compression Techniques

Overview & Theoretical Context

Deep learning-based methods have recently emerged as powerful tools for electrocardiogram (ECG) signal compression. Unlike traditional handcrafted compression approaches, deep neural networks learn hierarchical, task-optimized feature representations through data-driven training, as a result deep learning-based compression techniques can encode and reconstruct signals more effectively and can handle large amounts of data.

Techniques and Taxonomy

In the deep learning-based ECG compression technique a diverse range of neural network architectures have been proposed, each offering unique advantages in how temporal, morphological, and contextual features are captured and reconstructed. Convolutional Autoencoders (CAEs) are among the most widely used, leveraging convolutional layers to extract localized features from ECG waveforms and compress them into latent representations. [32] Their spatially aware structure makes them efficient and relatively lightweight for embedded implementations. Recurrent Autoencoders (RNN-AEs) and Long Short-Term Memory (LSTM)-based models are particularly useful in handling multi-beat ECG signals where rhythm continuity is critical.

Variational Autoencoders (VAEs) introduce a probabilistic element to encoding, allowing latent spaces to model distributions rather than fixed points. Some proposed methods also used Generative Adversarial Networks (GANs) to improve the visual and perceptual realism of reconstructed ECG signals.

Most importantly for deploying AI it is necessary to augment neural architectures with compression-aware training strategies such as quantization-aware training, pruning, low-rank matrix factorization, and knowledge distillation. It is also useful to incorporate rate-distortion optimization during training to balance compression ratio with signal fidelity dynamically.

Best Performing Techniques in Literature

Recent studies in cardiac signal compression demonstrate significant advances through neural network architecture. Yildirim and colleagues developed a 27-layer convolutional autoencoder system that achieves a compression ratio of 32.25 with only 2.73% average PRD when tested on the MIT-BIH arrhythmia database [37] . This architecture shows promise for mobile health applications due to its excellent balance between compression efficiency and signal preservation. In parallel work, Wang's team created an innovative autoencoder variant that systematically modifies data dimensionality throughout the compression process [38]. Their spindle-shaped architecture first expands then reduces feature dimensions, maintaining critical waveform information while achieving substantial size reduction. Their method achieved a high fixed compression ratio of 106.45 with an average PRD of 8.00%, demonstrating both efficient data reduction and quality-preserving ECG reconstruction. This technique demonstrated competitive compression on multiple datasets, but its deeper architecture introduces complexity that may hinder real-time deployment. In recent work on energy-efficient cardiac monitoring [39] authors introduced a compressive domain heartbeat classification system specifically optimized for wearable device constraints, demonstrating reduced computational requirements through modified sensing paradigms. By performing QRS detection and classification directly on compressed ECG signals without reconstruction, their approach achieved up to 90.00% accuracy at 40% compression ratio on the MIT-BIH database. The paper "New ECG Compression Method for Portable ECG Monitoring System Merged with Binary Convolutional Auto-Encoder and Residual Error Compensation"[40] presents a novel ECG compression technique that combines a Binary Convolutional Auto-Encoder (BCAE) with Residual Error Compensation (REC). This method achieved a remarkably high compression ratio of 117.33 while maintaining signal fidelity. It demonstrated an average PRD of 7.76%, with the best case reaching as low as 2.59%, indicating effective preservation of ECG signal quality. These results highlight the technique's suitability for real-time, portable ECG monitoring applications, where both high data reduction and diagnostic accuracy are essential. In the following

Table 2, we summarize comparatively best-performing deep learning-based ECG compression techniques and their strengths.

Table 2 Comparison of Deep Learning-Based ECG Compression Techniques

Technique	CR / PRD Range	Key Strength(s)	Reference
Spindle Convolutional Autoencoder (SCAE)	CR: 106.45 PRD: Average 8.00%	Extremely high compression ratio	Wang et al. (2019) [38]
Multi-Scale Deep Compressive Sensing Model	CR-2.5. as SNR: 37.66 dB PRD:1.55% to 2.48%	No signal reconstruction	Jing Hua et al. (2023) [39]
Deep Convolution-Based Encoding-Decoder Architecture	CR: 32.25 PRD: Average 2.73%	High compression ratio and low PRD	Yildirim et al., 2018 [37]
Binary Convolutional Auto-Encoder (BCAE) with Residual Error Compensation Network (RECN)	CR: 117.33 Percentage PRD: 7.76%	High compression ratio, Suitable for real-time systems	Shi, J.; Wang et al., 2022 [40]

2.4.3 Transform-Based Compression Techniques

Overview & Theoretical Context

Transform-based ECG compression techniques operate in the frequency or transformed domain, where the signal is represented by a set of coefficients that enable more efficient encoding. The primary objective of this transformation is energy compaction. In this technique most of the signal's energy is concentrated into a few dominant components, making it easier to compress the rest with minimal loss. Compression is typically achieved by discarding or quantizing coefficients that contribute minimally to the signal's perceptual quality or clinical value.

Techniques and Taxonomy

A variety of transform-based techniques are used in transform-based ECG signal compression; each of these techniques has specific advantages depending on signal characteristics and application constraints. The Discrete Wavelet Transform (DWT) is one of the most prominent methods due to its capacity for multi-resolution analysis, enabling it to capture both rapid transitions and slow-varying components within the same signal. DWT-based methods are often combined with entropy coders such as Huffman coding, Run-Length Encoding (RLE), or LZW to further reduce redundancy.

Discrete Cosine Transform (DCT) is another widely used technique, especially in block-based compression schemes. In typical implementations, DCT coefficients are quantized in a frequency-dependent manner to retain diagnostically relevant features while discarding noise or redundant data. Other dimensionality reduction approaches include Principal Component Analysis (PCA) and Singular Value Decomposition (SVD), which transform ECG signals into orthogonal subspaces for compact representation. These methods are effective for reducing inter-beat and intra-beat redundancy in multi-channel recordings. Recent advances have introduced cascaded transforms, such as DWT-DCT hybrids, and less conventional techniques like the Discrete Wave Atom Transform (DWAT). These cascaded transforms enhance sparsity for oscillatory signals.

Best Performing Techniques in Literature

We have reviewed recent publications of transform-based ECG compression techniques. Among the reviewed approaches, the Blaschke Unwinding Adaptive Fourier Decomposition (AFD) by Chunyu Tan et al. (2023) [41] emerged as one of the most efficient transform-based compressions. By iteratively decomposing ECG signals using analytic Blaschke functions, it achieved an impressive compression ratio 16.85 to 42.27 while maintaining PRD 0.81%–1.71% on the MIT-BIH arrhythmia database. This approach significantly outperforms traditional wavelet or cosine-based models. Table 3 enlisted some of the transform-based best-performing compression techniques and their strengths.

Table 3 Comparison of Transform-based ECG Compression Techniques

Technique	Reported CR / PRD Range	Key Strength(s)	Reference
Blaschke Unwinding Adaptive Fourier Decomposition (AFD) combined with Huffman encoding	CR 16.85 to 42.27 PRD: 0.81% to 1.71%	High compression efficiency and fast convergence	Chunyu Tan et al. (2019) [41]
1D Cohen + Daubechies-Feauveau 9/7 + Wavelet Transform	CR: 32.18 - Average PRD: 5.47 MIT-BIH	Automated and user-independent parameter optimization using LDI-PSO	Hardev Singh Pal et al. (2024) [42]
Q Wavelet Transform +Dead-Zone Quantizer + Run-Length Encoding	CR: ~10 to 45 PRD: ~1.2% to 6.7%	automate and improve TQWT parameter selection	Hardev Singh Pal, 2022 [43]
Modified Run-Length Encoding + Dead-Zone Quantization and Averaging	CR: 17.18 Average PRD: 3.92	Suitable for Holter monitoring applications	M.H. Kolekar, C.K. Jha, P. Kumar 2022 [44]

2.4.4 Hybrid Compression Techniques

Overview & Theoretical Context

Hybrid ECG compression techniques combine multiple compression strategies together, such as integrating transform-based methods with entropy coding or machine learning approaches, to optimize both compression efficiency and signal fidelity. Hybrid methods are especially beneficial in scenarios where relying on a single technique doesn't adequately balance high compression rates and accurate waveform reconstruction. Another advantage of using hybrid approaches is their adaptability; this technique allows us to customize the compression strategy for clinical or operational demands. However, these benefits typically come with increased complexity in algorithm design and the need of precise parameter optimization.

Techniques and Taxonomy

With a hybrid compression approach, the initial step often involves using the Discrete Wavelet Transform (DWT) to separate ECG signals into distinct frequency bands. After this, entropy encoding techniques such as Huffman coding, Run-Length Encoding (RLE), or Dictionary-based encoding are applied to efficiently compress these transformed coefficients. The main reason for using DWT is to group the critical ECG signal details into fewer, more compact coefficients, allowing entropy encoders to compress the data more effectively [45]. Another commonly adopted hybrid method is integrating machine learning techniques—such as neural networks or Support Vector Regression (SVR)with conventional signal processing approaches. The advantage of this combination is that it can improve ability to retain important clinical features of ECG signals, something that traditional methods alone sometimes struggle to achieve. Moreover, researchers have also combined methods like the Discrete Cosine Transform (DCT) with the DWT to further enhance the sparsity of ECG data, achieving even better compression performance. Techniques like K-means clustering can be implemented before the compression stage as well. These clustering methods group segments with similar waveforms, thus making the subsequent compression steps more effective by capitalizing on morphological similarities.

Best Performing Techniques in Literature

We have reviewed several recent ECG compression publications that used the Hybrid approach. Among this publication, the hybrid technique proposed by Zhang and their team achieved a good balance between PRD and CR [46] . They proposed a hybrid technique combining CSNet- (a ECG compression technique that uses Deep Learning) with Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) architecture. This technique achieved high reconstruction quality with compression ratios up to 90%, significantly outperforming traditional algorithms in both speed (up to 1810× faster) and clinical accuracy (PRD < 9%). Another recent study, "A novel ECG signal compression using wavelet and discrete anamorphic stretch transforms," [47] combined DAST (Discrete Anamorphic Stretch Transform) with DWT (Discrete Wavelet Transform) and RLE (Run-Length Encoding) . This method produced strong results, reaching a compression ratio of 27.81 while preserving the signal extremely accurately (the lowest PRD reported was 0%). In Table 4, we summarize some of the best-performing Hybrid ECG compression techniques and their strengths.

Table 4 Comparison of Hybrid ECG Compression Techniques with Reported CR and PRD

Technique	Reported CR / PRD	Key Strength(s)	Reference
CSNet (DL-based Compressed Sensing)	PRD: 1.88, SNR: 35.06 CR = 85% dB MITDB, NSTDB, INCART	Non-iterative and faster reconstruction	Zhang et al. (2021) [46]
Discrete Anamorphic Stretch Transform + Discrete Wavelet Transform + Run-Length Encoding	Compression Ratio (CR): up to 27.81 PRD: ranges from near 0 to ~5%	Superior CR compared to PRD	Thilagavathy R 2022 [47]
Discrete Cosine Transform + Fast Fourier Transform	CR range: 89.62 - 94.60 PRD range: 0.58% - 2.11%	High compression ratios achieved while maintaining low PRD values.	R. Mahajan and D. Bansal, 2014 [48]

2.4.5 Other Notable ECG Compression Techniques

Beyond these four above-mentioned ECG signal compression categories, there are many emerging compression techniques that have achieved higher CR and shown proven results on deploying real-time hardware. Compressed sensing (CS) approaches are another major technique used for ECG signal compression. In this process, ECG compression typically involves projecting the ECG signal into a lower-dimensional space using a sampling matrix, followed by reconstruction with sparsity-constrained algorithms. The effectiveness of this approach significantly depends on the choice of sparsifying basis, measurement matrix design, and reconstruction strategy. CS methods exploit inherent ECG signal sparsity in domains like wavelet, discrete cosine transform (DCT), or learned bases, thus reducing storage and transmission requirements and enabling sub-Nyquist sampling. These characteristics make CS particularly suitable for real-time and wearable applications with limited bandwidth and power resources [49] . There are several additional ECG compression approaches, though less conventional, that offer distinct benefits by addressing specific challenges through specialized data representation or advanced mathematical frameworks. Among these emerging methodologies, Symbolic Aggregate Approximation (SAX) is notable. This technique converts ECG signals into symbolic sequences by segmenting and summarizing the data. This technique is particularly useful in scenarios involving anomaly detection or long-term monitoring.

2.4.6 Comparison of ECG Compression Techniques

Among the four compression categories, the Lossless compression categories have a relatively low compression ratio; the highest CR can be up to 3.80, (Table 1) but this category is exceptionally useful where we need to maintain strict preservation of ECG signal integrity. These techniques are vital for clinical diagnostic accuracy and relatively less compound in the scope of real-time deployment and computational complexities. Due to its low complexity, many of the lossless techniques like ANS are suitable for the embedded platform deployment.

On the other hand, Deep Learning-Based Techniques can achieve exceptional compression efficiency. CR can often reach up to 130, (Table 2) but the PRD is also relatively high. These techniques have high adaptability through learned feature extraction, at the same time, the drawback of higher computational complexity and training overhead. These techniques are vulnerable to losing clinical features of the ECG signal after compression. Due to its complexity and the vulnerability of losing clinical features, deep learning-based compression techniques require real-time optimized architectures.

Transform-based techniques have effective frequency-domain representation and significant energy compaction capability. Through this compression technique, typically, a moderate to high compression ratio can be achieved, typically ranging from 10 to 45, (Table 3) with minimal distortion and outstanding approaches. In some cases, CR ratio can be achieved above 40. Unlike the deep learning-based compressor, the transformed-based techniques are moderately complex and shown proven capacities of real time deployment.

The unique advantage of the Hybrid Compression category is by using this technique; performance can be enhanced through a combination of multiple compression strategies, and high compression with the low distortion can be achieved through proper combinations of compression techniques. The main drawback of the Hybrid approach is its higher algorithmic complexity due to multi-step processing.

Lossless methods are crucial where diagnostic fidelity is paramount, offering moderate compression ratios. Deep Learning-based methods can achieve extremely high compression rates, ideal for big data and portable applications. Transform-based methods offer effective mid-range compression with robust diagnostic fidelity. Hybrid methods combine multiple strengths, delivering flexible, high-performance compression suited to diverse clinical needs. Choosing among these methods should consider clinical context, computational resources, desired compression ratio, allowable distortion, and real-time processing requirements.

2.5 Arrhythmia Classification Models and Evaluation Metrics

2.5.1 Traditional Arrhythmia Classification Models

Prior to the widespread adoption of deep learning, traditional machine learning algorithms were mainly used for arrhythmia classification. Traditional ML models, while conducting arrhythmia classification, rely heavily on manual feature extraction processes—these features are typically derived from temporal (e.g., RR intervals), frequency, and statistical properties of the ECG signal [50] . Commonly extracted features included time-domain descriptors like RR intervals, morphological parameters such as QRS complex width, P-wave duration, and frequency-domain features.

In the classical ML-based arrhythmia classification algorithms such as Decision Trees, Support Vector Machines (SVM), k-Nearest Neighbors (kNN), and Random Forests have proved substantial accurate results, especially when combined with well-engineered features. Among these, the most employed techniques were Support Vector Machines (SVM), which are effective for both binary and multiclass classification problems, particularly when the feature space is limited. K-Nearest Neighbors (kNN) is another widely used method, valued for its simplicity and interpretability, especially in small-scale ECG datasets [51].

ML approaches require extensive pre-processing steps, such as noise removal, QRS complex detection, and hand-crafted feature engineering. To refine model input, techniques such as Random Search Algorithms (RSA) and exhaustive grid search optimization are commonly employed. Decision Trees, especially with boosting techniques like AdaBoost.M2, achieved up to 99% accuracy in classifying normal vs. arrhythmic signals; SVMs, when combined with methods like wavelet transforms and genetic algorithms, reached accuracy as high as 99.66%. The ML classifiers remain robust and effective, particularly for small or structured datasets, they lack the adaptability and scalability of modern deep learning approaches, especially for raw and noisy ECG data.

2.5.2 Deep Learning-Based Arrhythmia Classification Models

Deep learning architecture is capable of automatically learning hierarchical features directly from raw or minimally pre-processed ECG signals. For this reason, in deep learning-based classification model there is no need for manual feature engineering. Moreover, this end-to-end learning paradigm has enabled significant improvements in classification performance, particularly on large-scale annotated ECG datasets [52].

In our literature review, we found that the Convolutional Neural Networks (CNNs) architecture has been predominantly employed in recent years for AI-driven cardiac arrhythmia detection systems. CNN architectures are well-suited for detecting spatial and morphological features in ECG signals. CNN has proven result in detecting morphological features such as QRS complexes, ST segments, and T-wave abnormalities. On the other hand, Recurrent Neural Networks (RNNs) demonstrate strong performance in learning long-range dependencies across ECG time-series signals. Long Short-Term Memory (LSTM) networks, a variant of RNNs, are especially effective at modelling long-term dependencies while avoiding vanishing gradient issues. This technique is ideal for arrhythmia detection over extended ECG recordings [53] .

More superior performance can be achieved by utilizing Hybrid models, for example integrating CNN and LSTM layers in a single architecture, this kind of model can capture both spatial and temporal feature. More recently, attention-based mechanisms and Transformer models have begun to emerge as promising alternatives for ECG sequence modelling, offering flexibility in capturing long-range dependencies without relying on sequential recurrence [54] .

In conclusion, the recent publication has proven that deep learning models have demonstrated significant improvements over traditional classifiers, particularly when sufficient labelled data is available. Their ability to generalize across patient populations and datasets makes them highly attractive for real-world deployment in automated ECG interpretation systems. While deep learning models significantly enhance classification accuracy, they require large, labelled datasets and substantial computational resources, which might limit their use in resource-constrained clinical environments.

2.5.3 Commonly Used Evaluation Metrics of Arrhythmia Classification

For fair and clinically meaningful comparison, arrhythmia classification models are evaluated under a set of standard performance metrics. These metrics allow a quantitative assessment of a model's ability to correctly distinguish between different cardiac rhythms. We need to deeply analyze several metrics particularly under conditions of class imbalance or diagnostic uncertainty. The following metrics are commonly used to evaluate the classification model.

Accuracy

Accuracy measures the proportion of total predictions that are correct. It provides a general sense of model performance but may be misleading in the presence of class imbalance. [55]

Supplementary metrics such as sensitivity, specificity, and F1-score provide more nuanced insights into model performance

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.9}$$

Here, TP, TN, FP, and FN represent True Positives, True Negatives, False Positives, and False Negatives, respectively.

Sensitivity (Recall)

Sensitivity, also known as recall, quantifies the model's ability to correctly identify actual positive cases — for example, correctly detecting arrhythmias when they are truly present. A high sensitivity is critical in clinical applications to minimize missed diagnoses. [55]

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{2.10}$$

Specificity

Specificity refers to the model's ability to correctly identify negative cases, such as normal sinus rhythm. High specificity ensures that healthy patients are not incorrectly flagged for arrhythmia, thereby reducing false alarms [55].

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{2.11}$$

Precision

Precision reflects the proportion of positive predictions that are correct. It is particularly important in situations where false positives carry significant clinical consequences [55].

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.12}$$

F1-Score

The F1-score is the harmonic means of precision and sensitivity. It provides a single score that balances both false positives and false negatives, making it especially useful in the presence of class imbalance [55].

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \tag{2.13}$$

Confusion Matrix

The confusion matrix offers a comprehensive view of model performance by presenting the counts of true positives, false positives, true negatives, and false negatives for each class. It enables detailed analysis of per-class errors and helps in diagnosing systematic misclassifications [56] .

Macro-AUC

For multiclass classification when dealing with class imbalance tasks Macro-AUC performance metric is commonly used. In simple terms, Macro-AUC measures how well a model can distinguish between multiple classes by averaging the individual AUC values calculated for each class. Specifically, it first calculates individual class AUC scores, then computes their arithmetic mean. This metric indicates the model's ability to correctly rank predictions for a specific class against all others [56].

Inference Time

Inference time refers to the computational duration required by a classification model to predict the class of a single ECG sample after the model has been trained. This duration is measured in milliseconds per sample (ms/sample). Inference time is calculated using the following relation:

$$\text{Inference Time} = \frac{\text{Total inference duration (ms)}}{\text{Number of samples tested}} \tag{2.14}$$

Lower inference times are generally desirable as they enable real-time clinical applications and faster decision-making processes [57] .

Model Size

Model size denotes the total storage space required to save a fully trained classification model, typically measured in kilobytes (KB) or megabytes (MB). It quantifies the memory footprint of the model when deployed and can be formally expressed as:

$$\text{Model Size} = \text{Storage required (KB or MB)} \tag{2.15}$$

Smaller model sizes are beneficial, particularly for deployment in resource-constrained or embedded healthcare systems, due to their reduced storage demands and lower complexity [58].

3 Methodology

3.1 Overview of Research Workflow

In this study we followed a structured experimental workflow designed to assess the impact of different ECG data compression techniques on the machine learning based ECG arrhythmia classification models. The primary goal is to determine which compression method best preserves clinically relevant features for accurate arrhythmia classification when the corresponding compressed ECG data is used in the classification model. The goal of the experiments is to identify the best compression technique by observing how the accuracy, model size, and inference time change with different compression techniques when applied in two representative machine learning-based classifiers: one is a CNN-LSTM-based arrhythmia classifier, and the other is a traditional RF + SVM-based ensemble model.

3.2 Dataset

In this experiment, we utilized the openly available MIT-BIH Arrhythmia Database. The main reason for using this database is that most of the contemporary publications used this database to implement a wide range of compression techniques. The MIT-BIH Arrhythmia Database is a foundational resource in cardiac electrophysiology and biomedical signal processing. Developed in the late 1970s by the BIH Arrhythmia Laboratory at Boston's Beth Israel Hospital and MIT, this database comprises 48 half-hour two-channel ambulatory ECG recordings collected from 47 subjects. Each recording has a sampling rate of 360 Hz per channel with an 11-bit resolution spanning a 10-mV range. A significant strength of this database is the presence of approximately 110,000 expert-labelled annotations, including different beat types and rhythm events [59]. Annotations are provided for each heartbeat, covering 20 distinct types, and are commonly grouped into five main classes following the AAMI recommendation: N (Normal), S (Supraventricular), V (Ventricular), F (Fusion), and Q (Unknown). Approximately 60% of the recordings were collected from hospital inpatients, ensuring realistic clinical variability. The dataset includes a wide demographic, which supports robust model training across age and gender differences. Among these recordings, 23 were randomly selected from over 4,000 Holter recordings to represent typical ECG patterns, while 25 records were intentionally chosen to include less common yet clinically significant arrhythmias, such as ventricular and supraventricular arrhythmias. Additionally, we used the MIT-BIH Supraventricular Arrhythmia Database for data augmentation purposes aimed at mitigating class imbalance during the training of our CNN-LSTM classification model.

Dataset Splitting

In the CNN-LSTM model we divided the dataset into mutually exclusive training and testing sets based on subject IDs to ensure that no recordings from the same individual appeared in both sets. We followed the same subject-wise split that is commonly used in literature for the MIT-BIH Arrhythmia Database. Specifically, records from one group of patients (DS1) were used exclusively for training, while a separate group (DS2) was reserved for testing. This subject-wise split effectively prevented data leakage by ensuring that the model was evaluated only on unseen patients.

For training and evaluating the RF-SVM ensemble model, a curated subset of records from the MIT-BIH Arrhythmia Database was used. As in this study we had to do a robust experiment and the RF-SVM ensemble model is computationally intensive, we used fewer records comparing the full MITBIH database. However, we have also verified the baseline model with the full number of records and found that the smaller number of records does not impact the accuracy. The dataset was split into training and testing sets using a stratified train-test split with a 70:30 ratio, ensuring a proportional representation of each arrhythmia class across both sets. To strictly prevent data leakage, segments from each ECG record were uniquely tracked, and no heartbeat segment from the same patient or recording appeared in both training and testing sets.

3.3 Preprocessing

Data preprocessing is a fundamental step for achieving optimal performance in machine learning models. In ECG classification, raw signals often contain considerable noise and artifacts that complicate direct interpretation by deep learning architectures. In this experiment we conducted different data preprocessing steps for different models, and the same preprocessing is also strictly maintained over all experimental setup to ensure compressed signal also pass through the same pipeline before applying to the classifier. This section outlines our comprehensive preprocessing pipeline implemented in Python. The preprocessing was mainly conducted using Python libraries such as SciPy, PyWavelets, and NumPy, etc.

3.3.1 Preprocessing Pipeline for Traditional RF-SVM Ensemble Model

We implemented a multi-stage preprocessing and feature engineering pipeline to prepare the ECG signals for the use of RF-SVM ensemble classification model. Raw ECG signals from the MIT-BIH Arrhythmia Database were loaded using the wfdb library, and only the primary signal lead was used for consistency. Baseline wandering was removed using a median filter with a kernel size of 201. The baseline-corrected signals were then denoised using a wavelet transform with the Daubechies 4 (db4) wavelet. A universal threshold, calculated based on the standard deviation of the highest-frequency coefficients, was applied using soft thresholding, and the denoised signal was reconstructed via inverse wavelet transformation. The ECG signals were segmented into fixed-duration windows (0.7 seconds) centered around annotated R-peaks to isolate individual heartbeats. Beat annotations were mapped into five classes as per the AAMI recommendations: Normal (N), Supraventricular (S), Ventricular (V), Fusion (F), and Unknown (Q). Segment-level features were extracted using the Time Series Feature Extraction Library (TSFEL). This TSFEL library can compute a comprehensive set of features spanning temporal, statistical, and spectral domains. Extracted features were standardized using z-score normalization, and missing values were imputed using mean imputation. To mitigate class imbalance in the training data, SMOTE (Synthetic Minority Over-sampling Technique) was applied using five nearest neighbors. Feature dimensionality was subsequently reduced using tree-based importance scores from a Random Forest classifier. We used the Select From Model method to retain features with above-median importance, this technique ensured only the most informative features were used in the final model training.

3.3.2 Preprocessing Pipeline for CNN-LSTM Hybrid Classifier

While preprocessing the ECG signal for the CNN-LSTM based classification model we utilized several systematic steps to ensure high-quality, standardized input data. We first denoised the raw ECG signals using a Daubechies 5 (db5) wavelet transform. This wavelet transform involved decomposing each signal into multiple levels of wavelet coefficients, applying a universal threshold to suppress high-frequency noise, and reconstructing the cleaned signal. After denoising, we normalized signals to have zero mean and unit variance. The cleaned signals were then segmented into fixed-length windows of 280 data points, each centered around annotated R-peaks. These segments provided temporally aligned beat-level samples for the model. Beat annotations were mapped into three clinically significant categories: Normal (N), Supraventricular (S), and Ventricular (V), for a simplified class structure. Finally, class labels were encoded into integer form and one-hot encoded for compatibility with categorical classification. Input samples were reshaped into a format of (280, 1) to match the expected input shape of the CNN-LSTM model. However, this preprocessing is not strictly universal to all scenarios of the experiment. To load differently compressed data and retain corresponding labels of the beat we moderately change the preprocessing steps and data loading logic. However, throughout the whole project we tried to ensure that the same preprocessing steps were utilized across all experiments.

3.4 Classification Models

3.4.1 RF-SVM Ensemble Classifier

In this study as the representative of the traditional machine learning based arrhythmia classifier, we used a voting-based ensemble model that combines the strengths of two well-established machine learning algorithms: Random Forest (RF) and Support Vector Machine (SVM). The Random Forest component consists of 100 decision trees trained using bootstrap aggregation, which offers resistance to noise and high variance in the data. The SVM component employs a radial basis function (RBF) kernel with probability estimation, allowing it to effectively handle complex, non-linear decision boundaries in high-dimensional feature spaces. To integrate these two classifiers, a Voting Classifier was implemented using both hard voting—where the predicted class is determined by majority vote—and soft voting—where class probabilities are averaged to make the final prediction. The ensemble used the SMOTE algorithm for tackling class imbalance and subsequently normalized with z-score standardization on the dataset before training. Features used for training were selected using a tree-based importance ranking method to reduce dimensionality and improve model efficiency. We exactly use the same baseline model that is trained and tested on uncompressed ECG data for the scenario of original training and compressed testing. We saved the trained baseline model, along with the corresponding scaler, feature selector, and label encoder, using the joblib library. Due to computational resource constraints, we configured the Random Forest classifier with 100 estimators and used an RBF-kernel SVM.

3.4.2 CNN-LSTM Classifier

To examine the impact of compression on the deep learning-based arrhythmia classification, we developed a hybrid CNN-LSTM (Convolutional Neural Network-Long Short-Term Memory) model. This model integrated both spatial and temporal feature extraction capabilities to efficiently analyze ECG signals. To address the class imbalance inherent in the MIT-BIH dataset, augmentation was performed by adding some S-class beats from the MIT-BIH Supraventricular Arrhythmia Database. Additionally, we also used oversampling via the SMOTE technique. We only classified three arrhythmia classes (Normal (N), Supraventricular (S), Ventricular (V)) in this classification model. Although our RF-SVM ensemble model classified 5 classes in the CNN-LSTM model we intentionally used 3 classes to keep the model simple. We also built a simple model architecture consisting of: Convolutional Layer: Initial Conv1D layer with 32 filters, kernel size of 7, ReLU activation, and 'same' padding. A MaxPooling1D layer with a pool size of 2 is used to reduce the dimensionality and computational complexity of extracted features. Two stacked LSTM layers were used, the first with 64 units configured to return sequences, and the second with 32 units, used to capture temporal dependencies and sequential characteristics of ECG data. A fully connected dense layer of 64 neurons with ReLU activation, followed by an output layer with 3 neurons employing a softmax activation function, is utilized, corresponding to the three arrhythmia classes (N, S, V). The model was compiled using the Adam optimizer and trained with categorical cross-entropy as the loss function.

3.5 Implemented ECG Compression Techniques

3.5.1 Lossless Compression Implementation

From our literature review, we found comparatively few publications are available on lossless compression compared to the publications of other compression techniques. In this thesis for implementing lossless compression, we followed the algorithm proposed by Banerjee and Singh[35] . In the implemented compression pipeline, a preprocessing step involving bandpass filtering is applied to the raw ECG signal using a second-order Butterworth filter with cutoff frequencies of 0.5 Hz and 40 Hz. Following filtering, the signal undergoes uniform quantization, where its amplitude values are linearly scaled and mapped to discrete integer levels within the range defined by an 11-bit resolution.

This lossless compression technique is based on second-order delta encoding. The compression process began with reading the raw ECG signal using the wfdb library, focusing on the primary channel. Each signal was first transformed using second-order differencing: the first-order difference was computed, followed by a second differencing step to highlight variations between consecutive signal derivatives. This produced a residual signal (d2) capturing fine-grained signal changes. To efficiently encode this second-order difference signal, the method computed an optimal threshold value (alpha) by analyzing the sparsity of large-magnitude changes in d2[35]. This threshold was used in conjunction with a dynamic integer coding scheme that grouped sequences of zeros and encoded them using run-length-like patterns. The encoding algorithm split the residual signal into three components: theta (mixed symbol stream), psi (run-length or placeholder indicators), and ksi (values requiring direct storage due to exceeding the threshold). Together, these formed the compressed representation. Decompression followed a reverse pipeline. The encoded components were reconstructed back into the second-order difference signal (phi), which was then reintegrated twice—first to recover the first-order difference using cumulative summation, and then to reconstruct the original signal. Furthermore, the algorithm preserved clinical annotations by saving the original sample indices and rhythm symbols from the database for each record.

3.5.2 Wavelet Based Compression Implementation

In this implementation, we closely follow the ECG compression methodology proposed in the paper "ECG Data Compression Using Modified Run-Length Encoding of Wavelet Coefficients for Holter Monitoring" by Kolekar et al., published in IRBM (2022) [44] . The technique incorporates hybrid elements due to the use of MRLE (a non-wavelet encoding method) and DZQ. Despite this, we categorized the paper as wavelet transform compression technique because the author emphasizes wavelet transform as the foundational step, and the hybrid steps (quantization, averaging, MRLE) are secondary optimizations. It is not a fully hybrid method like those combining wavelets with non-linear transforms or machine learning. The core idea of this method is to leverage the energy compaction capability of discrete wavelet transform (DWT) and eliminate redundancy using dead-zone quantization (DZQ) followed by a modified run-length encoding (MRLE) scheme [44] .

First, each ECG record is decomposed using a 5-level Daubechies-2 (db2) wavelet transform. The resulting wavelet coefficients are then quantized using DZQ with level-dependent thresholds—an enhancement we introduced to adjust quantization aggressiveness across sub bands based on signal importance. Next, in accordance with the original paper, we compute the energy packing efficiency (EPE) of each sub band and replace detail coefficients with their average value if EPE falls below a specified threshold, thereby enhancing compressibility. To maintain waveform fidelity, we skip averaging and rounding on the approximation coefficients. The MRLE technique is applied to the processed coefficients, adding a slight offset to distinguish single occurrences and multiplying values by 10 for compact storage. As a minor improvement, we added a dynamic type of selection mechanism that uses int8 encoding whenever feasible to reduce storage overhead. Prior to encoding, detail coefficients are rounded to integers to reduce entropy and improve compression effectiveness. The signal is reconstructed using inverse DWT after inverse quantization to evaluate reconstruction quality via PRD. In addition, beat annotations (R-peaks and symbols) are extracted from the record annotations and stored separately. Collectively, our implementation faithfully captures all the major principles of the original method while introducing a few empirically driven modifications to enhance compression ratio without compromising signal quality.

3.5.3 Deep Learning Based Compression Implementation

In this study, we implemented an autoencoder-based ECG compression technique based on the methodology presented in the paper titled "New ECG Compression Method for Portable ECG Monitoring System Merged with Binary Convolutional Auto-Encoder and Residual Error Compensation" by Jiguang Shi et al. (2022) [40] . The compression approach utilizes a Binary Convolutional Auto-Encoder (BCAE) combined with a Residual Error Compensation Network (RECN). The BCAE effectively compresses ECG heartbeats segmented from the MIT-BIH dataset into concise binary representations. Specifically, each heartbeat consisting of 320 samples is processed by the encoder, generating compact 20-bit binary codes through convolutional layers, max-pooling operations, and a custom binary activation function. Subsequently, the decoder reconstructs signals from these binary codes using transposed convolutional and up-sampling layers. To improve reconstruction accuracy further, the RECN—structured utilized as a Multi-Layer Perceptron (MLP)—predicts and compensates for residual errors between original and reconstructed signals. Training was performed in two stages: initially optimizing the BCAE to minimize reconstruction loss and then training the RECN on the computed residual errors. This implementation was developed using TensorFlow and Keras, demonstrating high-quality signal reconstruction suitable for subsequent arrhythmia classification tasks. The final compressed and reconstructed ECG signals, along with the exact annotations (beat labels), were explicitly saved in structured MATLAB-compatible (.mat) files for subsequent arrhythmia classification tasks.

3.6 Evaluation Criteria and Experimental Scenarios

To determine the effect of different compressions on the arrhythmia classification models, we calculated Accuracy, Precision, Recall, F1-Score, Macro AUC, Inference time and the binary model size for each representative classification model across all experimental scenarios. Three experimental scenarios were observed:

In Scenario 1, we used original uncompressed ECG data for both training and test dataset in two representative classification models and set each model as the baseline model.

In Scenario 2, we used the original data as the training dataset but as the test data, we used compressed data. For Scenario 2, we used the saved model from Scenario 1.

In Scenario 3, we used compressed data for both the training dataset and test dataset. As both training and testing data were compressed in Scenario 3, we trained the model again but used the same model architecture.

In all these scenarios, the same preprocessing steps were strictly maintained. We explicitly ensured that there was no label mismatch and preserved the integrity of the evaluation process. By comparing outcomes across scenarios, we aim to identify the optimal compression method balancing accuracy and efficiency for ECG-based arrhythmia detection.

3.7 Methodological Justification for Model Selection

To comprehensively evaluate the impact of ECG signal compression on arrhythmia classification from both deep learning and traditional machine learning perspectives, we strategically selected two classification models: a CNN-LSTM and an RF+SVM ensemble. The CNN-LSTM model was specifically chosen due to its proven capability in effectively handling sequential data, particularly time-series signals such as ECG. We also observed, CNN-LSTM architectures have consistently demonstrated strong performance in recent literature, therefore we selected this model for evaluating compression impacts.

The RF+SVM ensemble was selected to represent traditional machine learning approaches due to its interpretability, robustness, and generalization capability. This dual-model approach thus provides critical insights into the distinct sensitivities and robustness characteristics of traditional machine learning versus deep learning methods in the context of ECG compression.

4 Results and Observations

4.1 Compression Metrics Achieved in the Lossless Technique

In the lossless implementation, we achieved an average CR of 2.29. The decompressed ECG data were explicitly saved as NumPy arrays, and all original annotations were preserved in JSON format for subsequent analysis. However, the compression ratio we achieved was lower than the compression ratio (CR = 3.82) reported in the original paper for the MIT-BIH dataset. This difference may be due to subtle implementation details not explicitly described in the original methodology, differences in header bit assumptions, or dataset handling nuances.

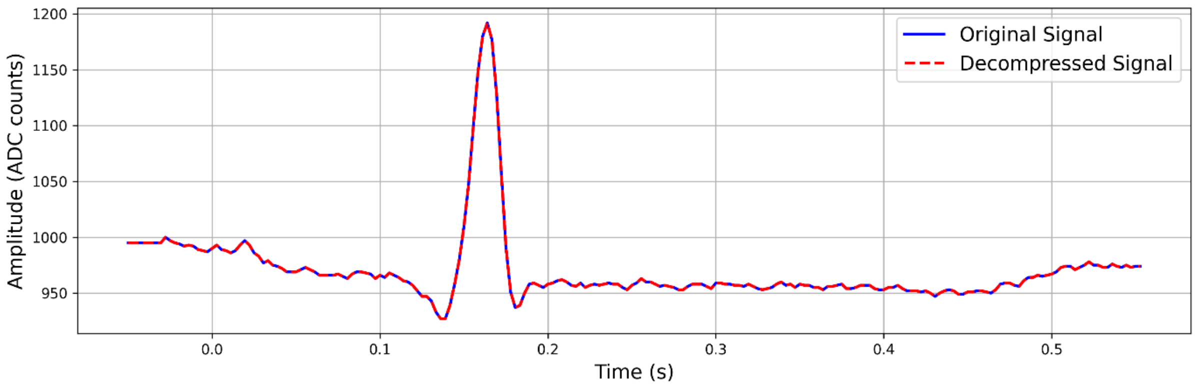

Figure 5 Comparison of a single beat before and after applying lossless compression.

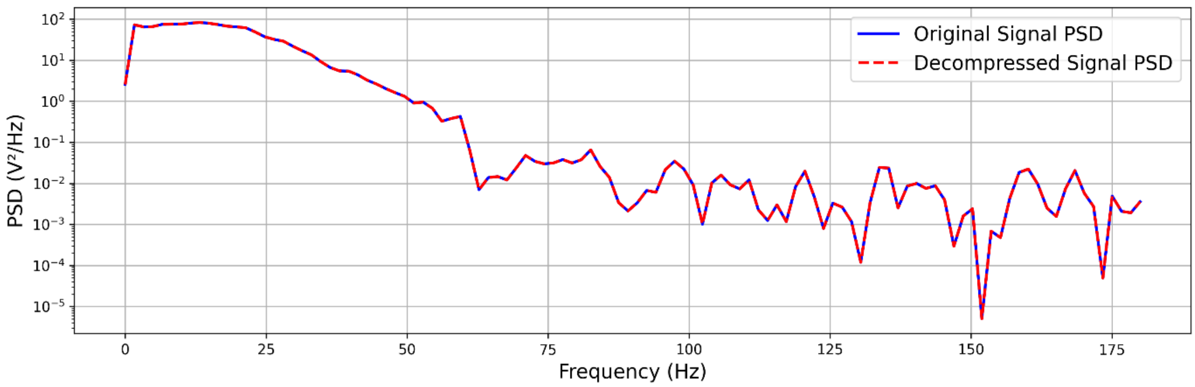

Figure 6 Power Spectral Density (PSD) comparison of the raw and lossless compressed ECG signals.

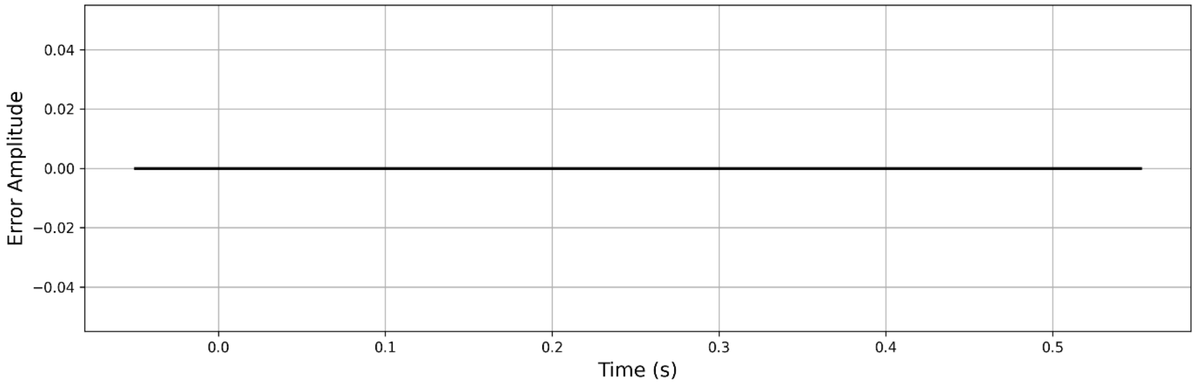

Figure 7 Reconstruction error between the uncompressed and decompressed ECG signals after applying lossless compression.

4.2 Compression Metrics Achieved in the Wavelet-based Technique

In our implementation, the wavelet-based compression achieved an overall Compression Ratio (CR) of 10.67, and the reconstructed signals showed an average PRD of 4.64%. All compressed signals, annotations, and reconstructed signals were saved appropriately in .pkl and .mat file formats to ensure reproducibility and subsequent analysis in the classification models. In the original study by Kolekar et al. (2022)[44] , the wavelet-based ECG compression method achieved an average Compression Ratio (CR) of 17.18 and a Percentage Root Mean Square Difference (PRD) of 3.92% using all 48 MIT-BIH ECG records and MATLAB implementation. The compression metrics deviation in our implementation is possibly attributed to differences in wavelet parameterization, platform-specific behaviors, and scope limitations.

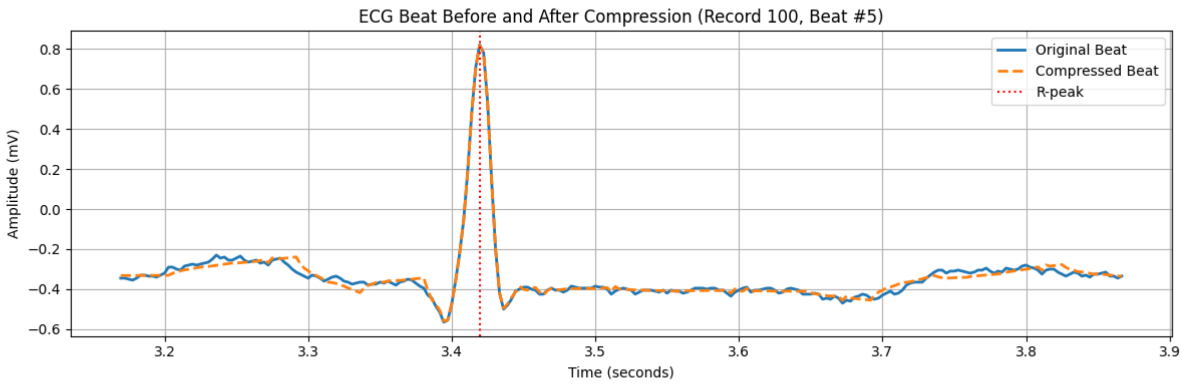

Figure 8 Comparison of a single ECG beat from MIT-BIH record 100 before and after applying wavelet-based compression.

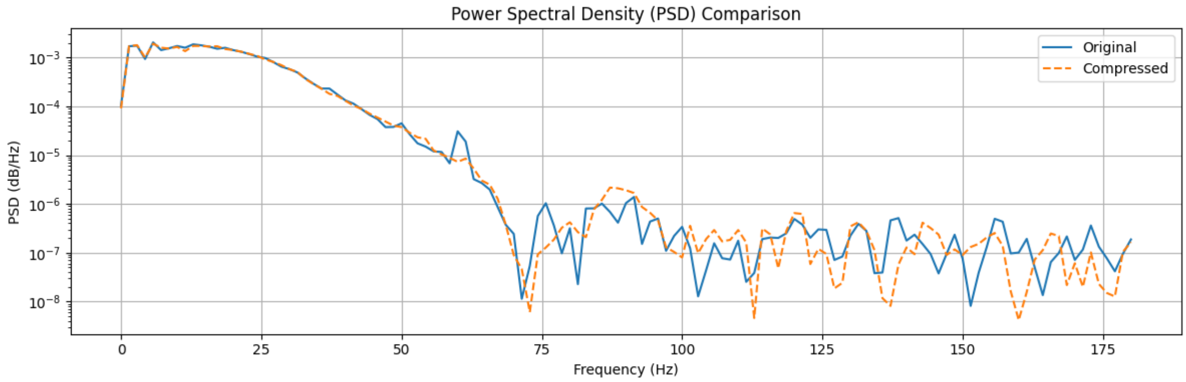

Figure 9 Power Spectral Density comparison between the uncompressed and the wavelet compressed ECG signal.

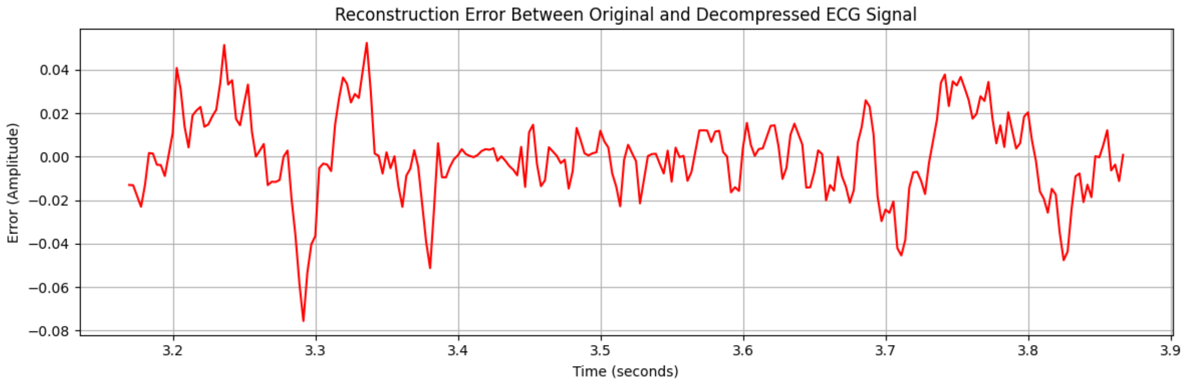

Figure 10 Visualization of reconstruction error between original and decompressed ECG signals after wavelet compression.

4.3 Compression Metrics Achieved in the Deep Learning-Based Technique

The implemented deep learning-based ECG compression achieved an overall CR of 117.33 and a PRD of overall 10% across all 48 records in the MIT-BIH Arrhythmia Database. These results closely align with the findings of the original study by Shi et al. (2022) [40] , which reported a CR of 117.33 and an average PRD=7.76%.

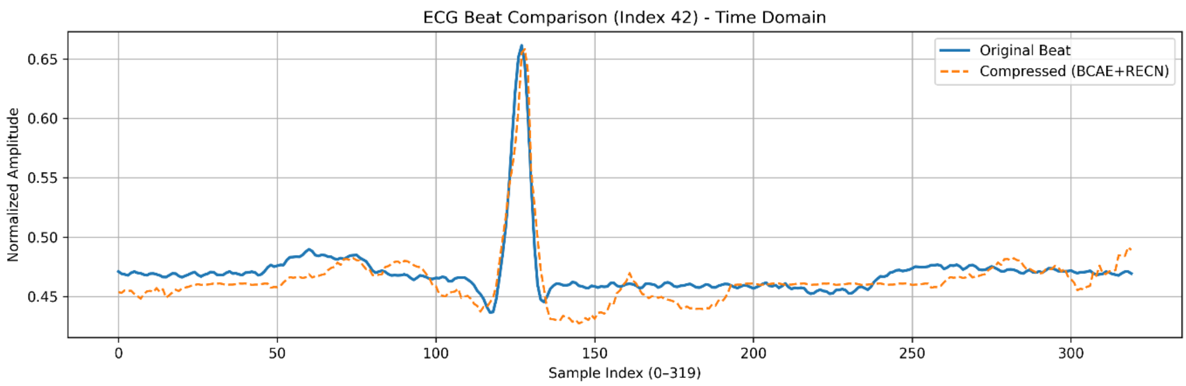

Figure 11 Comparison of a single ECG beat before and after applying auto encoder compression.

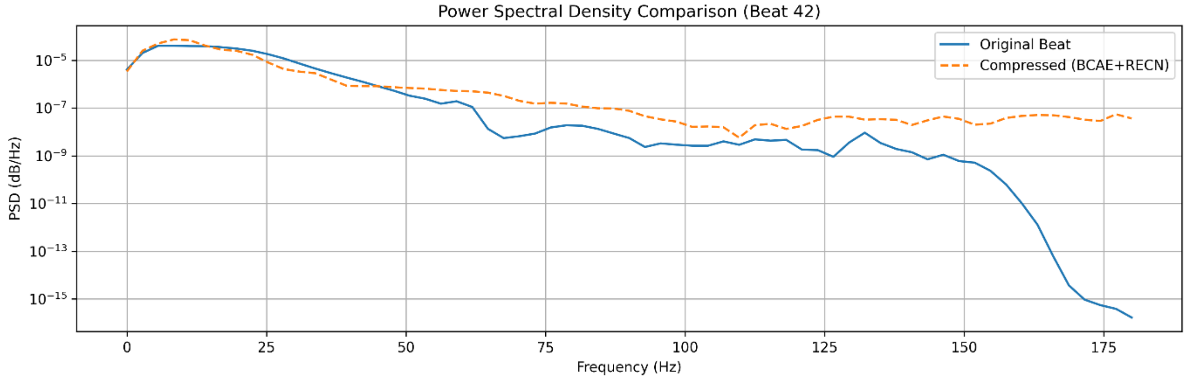

Figure 12 Power Spectral Density comparison between the uncompressed and the autoencoder compressed ECG signals for a selected beat.

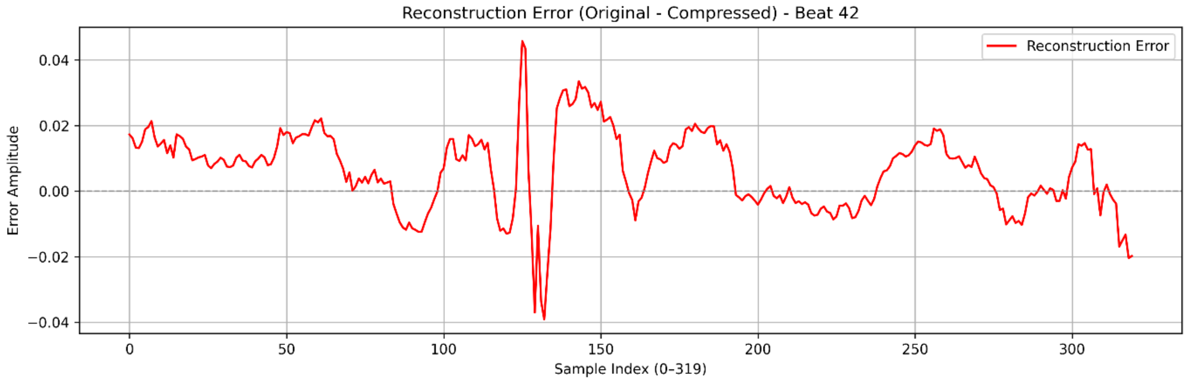

Figure 13 Visualization of reconstruction error between original and decompressed ECG signals after applying autoencoder.

4.4 Summary of Results from the Impact of ECG Data Compression on Classification Models

While using different compressed data obtained from Lossless, Wavelet, and Autoencoder compression on the classification models we observed performance changes in the classification models with different training and test data combinations (compressed or uncompressed) and also with the type of compression technique used.

After conducting experiments across all scenarios, we present the key metrics essential to understanding the impact of ECG compression on the classification models. Table 5 summarizes baseline model performance metrics using original training and original testing datasets.

Table 5 Baseline Model Accuracy Original Training Original Testing

Compression Type	RF+SVM Ensemble	CNN LSTM	AVG.
Macro-AUC	0.921	0.847	0.884
Weighted Avg Precision	0.88	0.90	0.89
Weighted Avg F1-score	0.85	0.86	0.855
Inference Time	1.695 ms	0.28 ms	0.987 ms
Model Size	105.92 MB	0.50 MB	53.21 MB

Our arrhythmia classification models were not intended to be state-of-the-art solutions. We mainly focused on building classification models as tools to evaluate the impact of ECG signal compression on model performance; consequently, we observed a moderate performance in our baseline model. From the baseline performance results presented in Table 5, we observed that the RF+SVM ensemble model achieved a higher Macro-AUC (0.921) compared to the CNN LSTM model (0.847). However, the CNN LSTM model has a significantly better inference speed (0.28 ms) and was substantially smaller in model size (0.50 MB) compared to the RF+SVM ensemble (1.695 ms and 105.92 MB, respectively). CNN LSTM model provided slightly better weighted average precision (0.90) and marginally superior weighted average F1-score (0.86) relative to the RF+SVM ensemble (0.88 precision and 0.85 F1-score).

Table 6 shows the performance metrics for the models when it trained on original data and tested on different compressed data.

Table 6 Performance metrics comparison for models trained on original data and tested on compressed data across three compression methods (Lossless, Wavelet, Autoencoder). Metrics include Macro-AUC, Weighted F1-score, Model Size (MB), and Inference Time

Compression Type	Model	Macro-AUC	Weighted F1-score	Model Size (MB)	Inference Time (ms)
Lossless	RF+SVM Ensemble	Nan	0.710	105.92	1.382
	CNN LSTM	0.8470	0.8575	0.50	0.310
	Average	0.8470	0.783	53.21	0.846
Wavelet	RF+SVM Ensemble	Nan	0.510	105.92	1.486
	CNN LSTM	0.6935	0.8286	0.50	0.310
	Average	0.6935	0.669	53.21	0.898
Autoencoder	RF+SVM Ensemble	Nan	0.760	105.92	1.617
	CNN LSTM	0.6971	0.8185	0.50	0.270
	Average	0.6971	0.789	53.21	0.943

Based on the results shown in Table 6, we observed that the RF+SVM ensemble model consistently failed to produce a valid Macro-AUC under all compression techniques when the model is trained on the original data but tested on the compressed data. The absence (NaN) of valid Macro-AUC results from the RF+SVM ensemble indicates an inability of this model to generalize to compressed ECG signals under the conditions of original training data and compressed test data. CNN LSTM model also performed less than the baseline model but it successfully maintained reasonable Macro-AUC values (ranging from 0.6935 to 0.8470) and also exhibited stable weighted F1-scores (between 0.8185 and 0.8575). The inference times did not change a lot for the CNN LSTM model, but the RF+SVM ensemble model showed inference times ranging between 1.382 and 1.617 ms.

Table 7 demonstrate the performance metrics of the classification models on three compressed data categories when the model is trained and tested both on compressed ECG data.

Table 7 Performance metrics comparison for models trained and tested on compressed data across three compression methods (Lossless, Wavelet, Autoencoder). Metrics include Macro-AUC, Weighted F1-score, Model Size (MB), and Inference Time per sample (MS).

Compression Type	Model	Macro-AUC	Weighted F1-score	Model Size (MB)	Inference Time (ms)
Lossless	RF+SVM Ensemble	0.919	0.840	104.31	1.680
	CNN LSTM	0.7340	0.8526	0.50	0.270
	Average	0.826	0.8463	52.40	0.975
Wavelet	RF+SVM Ensemble	0.920	0.840	106.49	1.5721
	CNN LSTM	0.7043	0.791	0.50	0.263
	Average	0.812	0.815	53.495	0.917
Autoencoder	RF+SVM Ensemble	0.946	0.940	97.42	0.714
	CNN LSTM	0.6255	0.6901	0.50	0.310
	Average	0.785	0.815	48.96	0.512

From Table 7, we found the RF+SVM ensemble model consistently retained higher Macro-AUC scores (ranging from 0.919 to 0.946) and weighted F1-scores (0.840 to 0.940) after applying different compressed signal as the training data and test data. The CNN LSTM model again showed poor performance compared to the baseline model. Inference times of the both model flactutaed slightly more than the scenarios when the models were trained on original data and tested on the compressed data.

4.5 Observations

4.5.1 Impact of Compression on Model Size

Our experiments clearly indicate that the application of ECG signal compression, regardless of the compression method—lossless, wavelet-based, or autoencoder-based—generally has no observable impact on the size of the deep learning-based classification models, specifically for the CNN-LSTM model. For the CNN-LSTM model, the model size remained consistent, 0.50 MB across different training and testing scenarios of all compression techniques. However, in the case of the RF+SVM ensemble model, we observed slight variations in model size across different compression techniques, specifically when the model is trained and tested on the compressed data. The model size of the RF+SVM ensemble model was reduced notably when the autoencoder compressed training and test data was used model size dropped to 97.42 MB (Table 7) from the 105.92 MB baseline model (Table 5), suggesting that the autoencoder may effectively simplify feature extraction. From a scientific standpoint, the general consistency in the model size of the deep learning-based classification model is logical, as the size of these classification models is fundamentally determined by their architecture, parameter count, and computational complexity. However, our experiment found that the model size of the traditional Machine learning-based models can be reduced with the ECG data compression.

In this study, we only calculated the binary file size of the saved model. Because ECG compression alters input data only, the network weights remain identical; hence, flash/SRAM footprints are also constant across all compression settings.

4.5.2 Impact of Compression on Class Feature Recognition

In the case of the traditional RF+SVM ensemble classifiers, we observed the appearance of NaN (Not a Number) values in all the scenarios (Table 6) when the classifier was trained on original ECG data but tested on compressed ECG data. The primary reason behind this issue is the classifier's complete inability to recognize any or some of the small classes such as 'S', 'V', 'F', and 'Q' depending on the compression techniques. This implies that compression can transform the original features of the ECG signals significantly enough to distort or eliminate vital class-specific characteristics that are essential for the RF+SVM ensemble model when the model is trained on original data but tested on the compressed data. Specifically, the absence of true positive predictions for certain classes directly resulted in an AUC of zero, thus producing NaN values in the Macro-AUC calculation. However, this phenomenon was not noticed in the CNN-LSTM model in the same scenario where the training data is original, but the test data is compressed. Although in our experiment the CNN-LSTM model predicted three classes, it recognised each of them across all compression techniques and training test combination. In contrast, the RF-SVM model failed entirely to identify beats from certain classes when trained on original data and tested on compressed signals, resulting in no correct predictions for these classes.This finding underscores a critical limitation of traditional machine learning models and suggest that it is required either to use same compressed or uncompressed data on both training or test dataset or to adopt more robust feature extraction methods resilient to compression-induced changes.

4.5.3 Impact of Compression on Inference Time

The traditional RF+SVM ensemble model, when trained on original ECG data and tested on compressed ECG data, generally showed slightly reduced inference times from the baseline model. Compared to the baseline inference time 1.695 ms (Table-5) when we used Lossless, Wavelet and Autoencoder compressed test data with original training data we found inference time 1.382 ms, 1.486 ms and 1.617 ms respectively (Table-6). The CNN-LSTM model exhibited stable inference times with minimal variation, ranging from 0.27 ms (Autoencoder), to 0.31 ms (Wavelet).

For Compressed training & Compressed test scenario, from the Table-7 we find in case of the RF+SVM ensemble model inference times decreased slightly for Lossless compression (1.680 ms), Wavelet compression reduced inference time (1.5721 ms) and the Autoencoder compression notably reduced inference time (0.714 ms), indicating a significantly more computationally efficient feature representation. Meanwhile, the CNN-LSTM model maintained consistently stable inference times across compression methods, varying slightly between 0.263 ms (Wavelet) to 0.31 ms (Autoencoder).

As noted previously, the RF+SVM ensemble model is particularly sensitive to changes in the feature space because compression typically transforms or simplifies the input data, potentially reducing computational complexity during decoding or preprocessing; thus, inference times may decrease slightly. Conversely, deep learning-based classifiers directly learn complex feature representations through internal adaptive mechanisms within their neural network architectures. Their inference times remain consistently stable because internal processing of compressed data does not significantly alter their computational complexity.

4.5.4 Impact of Compression on Accuracy

From Table 6, we can observe that when the CNN-LSTM model is trained on original ECG signals and tested on compressed signals, in case of lossless compression, the CNN-LSTM model maintained stable performance with a Macro AUC of 0.847, identical to the baseline. However, wavelet compression significantly reduced the Macro AUC to 0.6935. Autoencoder compression also reduced Macro AUC performance to 0.6971, with a corresponding drop in Weighted Average F1-score to 0.8185, and this was expected as the

PRD of the autoencoder compression was quite high. In scenarios where both training and testing data were compressed ECG signals, the CNN-LSTM model demonstrated reduced performance from the baseline model. In this case, the model gained Macro AUC 0.734 for lossless, Macro AUC 0.7043 for wavelet compression, and the lowest Macro AUC 0.6255 for autoencoder compressed (Table 7) substantial decrease in Weighted Average F1-score is also observed across all scenarios.

For the RF+SVM ensemble model, testing compressed data after training on original ECG data led to severe performance issues, as indicated by the inability to compute valid Macro AUC (NaN) across all compression techniques. Nevertheless, Weighted Average F1-score varied notably under this scenario, with values of 0.71 for lossless compression, 0.51 for wavelet compression, and an improved 0.76 for autoencoder compression, revealing substantial class-specific prediction issues (Table 6).

When the RF+SVM model was trained and tested on compressed data, the results improved substantially. Lossless and wavelet compression demonstrated stable Macro AUC performances 0.919 and 0.920, respectively (Table 7) along with consistent Weighted Average F1-scores at 0.840. Autoencoder compression provided the best performance improvement, with a notably high Macro AUC of 0.946 and the highest Weighted Average F1-score of 0.940, clearly demonstrating its suitability for the RF+SVM ensemble model.

Average Evaluation Metrics Across Experimental Scenarios

Table 8 summarizes accuracy (Macro-AUC), model size, and inference time on average of the two classification models in different compression techniques and different training-test combinations.

Table 8 Summary of AVG Evaluation Metrics over two Classification models

CT Type	Baseline Scenario			Original Training Data Compressed Test Data			Compressed Training and Test Data		
Evaluation Metrics	Macro -AUC	Model Size (MB)	Inference Time / Sample (ms)	Macro-AUC	Model Size (MB)	Inference Time / Sample (ms)	Macro-AUC	Model Size (MB)	Inference Time / Sample (ms)
Lossless	0.884	53.21	0.987	0.8470	53.21	0.846	0.826	52.40	0.975
Wavelet				0.6935	53.21	0.898	0.812	53.49	0.917
Autoencoder				0.6971	53.21	0.943	0.785	48.96	0.512

The average Macro-AUC of the two-baseline classification model was 0.884. Model Size and Inference Time were 53.21 MB and 0.987 ms/sample, respectively.

On average, for the scenario of original Training and Compressed Test Data, Macro-AUC is comparatively higher for the Lossless compressed test data. Model size remains identical to the baseline average model size. A slight variation in inference time (marginally improved) is observed from the baseline model.

For the compressed Training and compressed Test Data scenario, the none of the average Macro-AUC reach the baseline performance. Model sizes fluctuate slightly: Lossless (52.40 MB), Wavelet (53.49 MB), and Autoencoder noticeably smaller (48.96 MB). Inference Time notably improved, particularly for autoencoder compression (0.512 ms/sample), indicating substantial computational efficiency gains.

RF-SVM Ensemble Model Performance Evaluation using ROC-AUC Curves

The ROC curves for the baseline scenario of the RF-SVM ensemble model indicate strong classification performance across multiple classes (Figure 14). The model shows a good performance when the autoencoder compressed training data and the autoencoder compressed test data (Figure 15) is used. In this scenario, the ROC-AUC scores improved notably in all classes except the S class. This phenomenon suggests that the autoencoder compression significantly aids model generalization or discriminative power for certain classes. The RF SVM ensemble model performed worst when the original Data is used as the Training Data and Wavelet Compressed data is used as the Test data (Figure 16). In this scenario, the ROC-AUC scores drastically decrease, near a random guessing level. This highlights the critical role of matching training and test data preprocessing strategies for achieving optimal predictive performance.

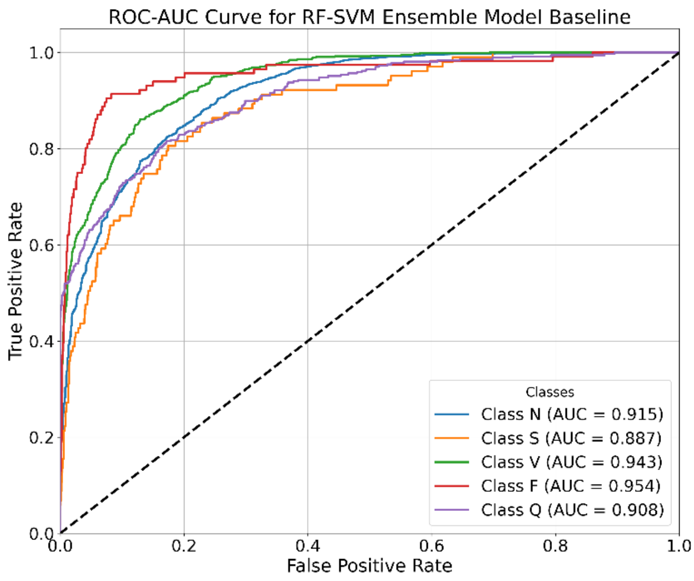

Figure 14 ROC-AUC curves for RF-SVM Ensemble Model trained and tested on original ECG data, showing strong multi-class classification performance.

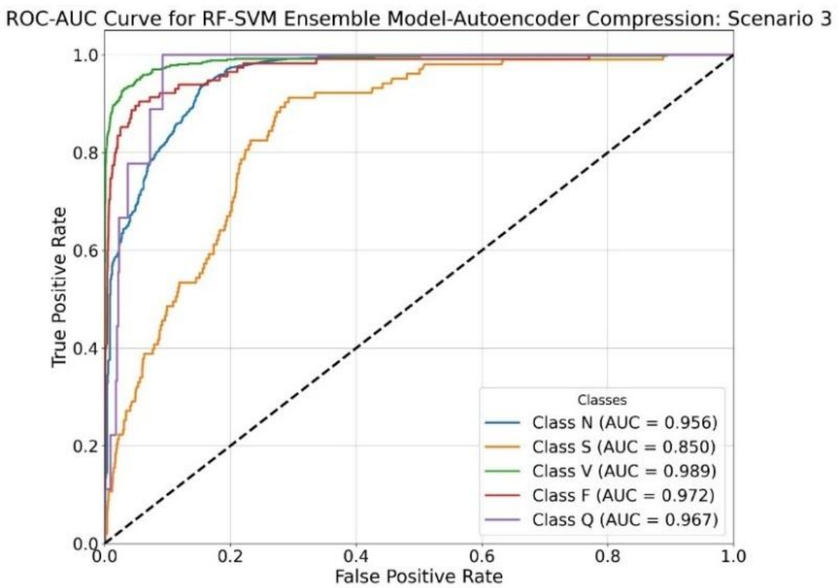

Figure 15 ROC-AUC curves for RF-SVM Ensemble Model trained and tested on Autoencoder-compressed ECG data, indicating improved or maintained classification effectiveness.

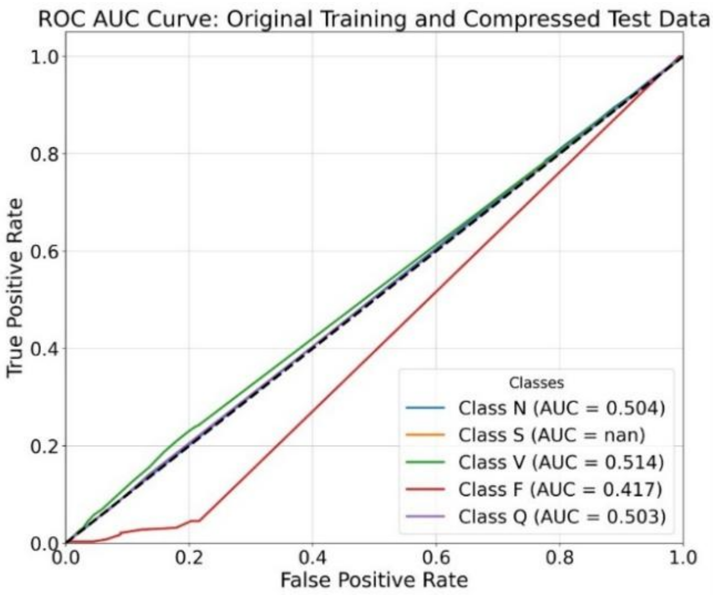

Figure 16 ROC-AUC curves for RF-SVM Ensemble Model trained on original ECG data but tested on wavelet-compressed ECG data, illustrating significant performance deterioration.

CNN-LSTM Model Performance Evaluation using ROC-AUC Curves

The ROC curves (Figure 17) of the baseline CNN-LSTM model (trained and tested on the original ECG data) performs best at classifying class "V," with an Area Under the Curve (AUC) of 0.957. While using different compressed data in this model for different training and test combinations we find the model reasonably performed best for the scenario when the original training data and compressed lossless test data is used. We found an identical ROC curves (Figure 18) like the baseline model for this scenario. The worst performing scenario observed when the model is trained and tested on the autoencoder compressed ECG signal(Figure 19) . In this scenario all the class performances has reduced significatly from the baseline performance. Specifically for Class S the AUC = 0.493 that performed even lower than the random performance.

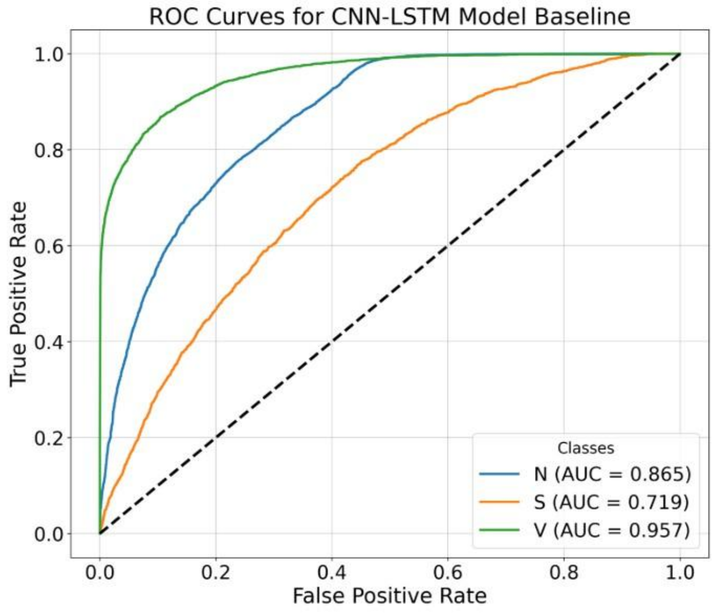

Figure 17 ROC-AUC curves for CNN-LSTM Model trained and tested on original ECG data, showing strong multi-class classification performance.

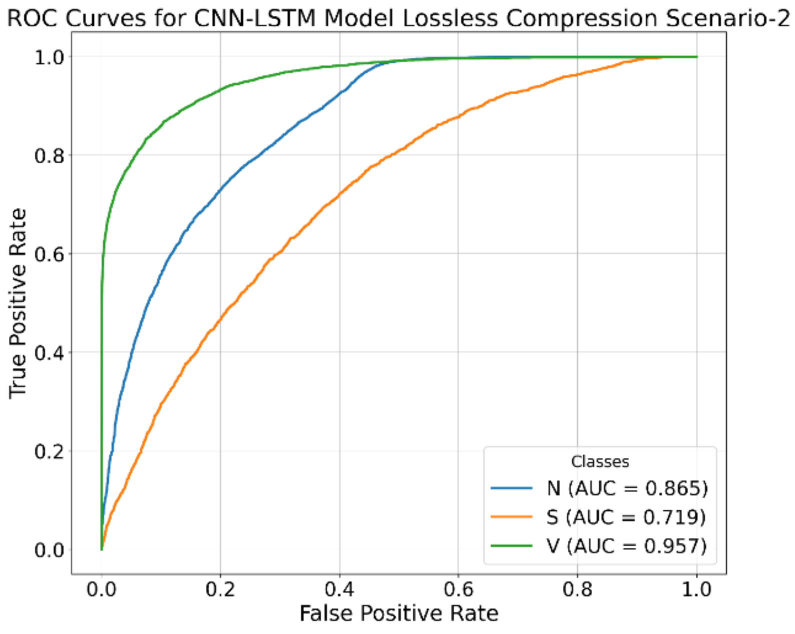

Figure 18 ROC-AUC curves for the CNN-LSTM Model trained on original ECG data and tested on lossless compressed ECG data. (This curve is identical to the CNN-LSTM baseline curve)

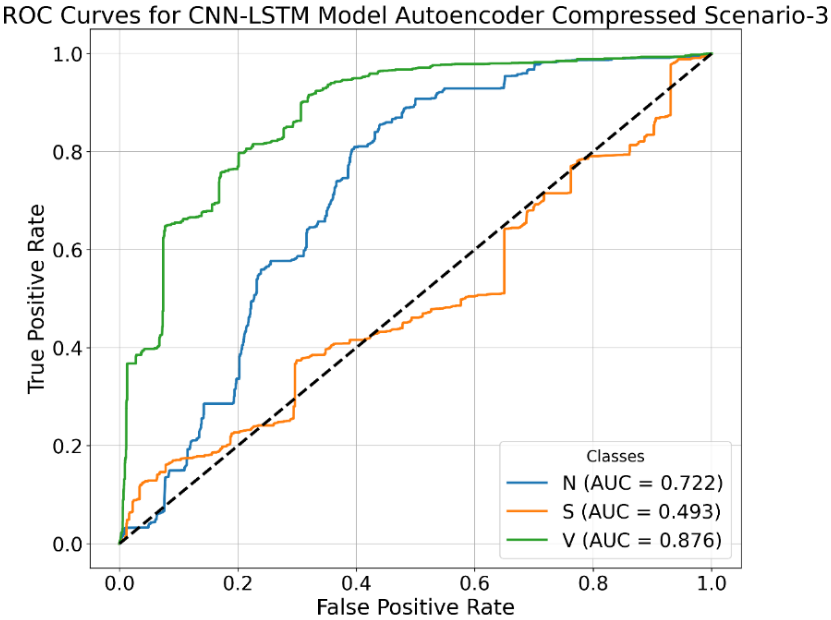

Figure 19 ROC-AUC curves for CNN-LSTM Model trained and tested on autoencoder compressed ECG data.

4.6 Core Findings

After examining all the metrics and visual results, we can summarize as follows:

1. ECG signal compression techniques have negligible or no impact on the model size of the CNN-LSTM based classification model.

2. Traditional machine learning classifiers RF+SVM ensemble model fail to recognize specific arrhythmia classes when trained on original ECG data but tested on different compressed ECG data

3. Deep learning model CNN-LSTM demonstrate good performance if the model is trained on the original ECG Data and tested on the lossless compressed ECG data. But the performance of this model reduced significantly if it is trained and tested on autoencoder compressed ECG data.

4. The inference time for traditional RF+SVM ensemble classifiers slightly varied in all cases due to the indirect impact of compression. The autoencoder-based compression techniques can even reduce inference time of the RF+SVM ensemble significantly.

5. Autoencoder-based compression significantly improves the accuracy of the traditional RF+SVM ensemble model when the model is trained and tested on compressed data, making it a valuable strategy for enhancing traditional classifier performance. This compression also reduced the model size and the inference time of the model.

6. The performance of the CNN-LSTM model reduced in all scenarios where compressed ECG data is used either test or training data, but the model did not completely fail to identify any class like the RF+SVM ensemble model.

These core findings collectively guide the informed integration of ECG signal compression techniques into AI-driven arrhythmia classification frameworks, ensuring optimized model accuracy, computational efficiency, and clinical reliability.

5 Discussion and Conclusion

5.1 Limitations of the Study

Although this thesis thoroughly investigated how different ECG signal compression techniques influence the arrhythmia classification models, we would like to state certain constraints of our study for transparency and future development of this project. The most notable limitation of this study is the dataset inadequacy. We selected only one dataset, the MIT-BIH Arrhythmia Database, while well-established and frequently utilized, but this dataset only contains 48 short-duration ECG recordings. Real-world ECG data typically exhibit far greater complexity; for example, the PhysioNet 2021 challenge datasets contain a broader spectrum of arrhythmias, varying signal qualities, and diverse patient characteristics. Because of this dataset limitations, the conclusions drawn from our experiments might not be universally generalizable to all clinical scenarios or patient populations.

Another specific limitation emerged from our RF+SVM ensemble model experiments, where the model consistently produced NaN values for Macro-AUC when testing compressed data on a model trained with original ECG signals. This highlighted a critical vulnerability of traditional classifiers to compression-induced feature alterations. Unfortunately, given the scope and timeline, we did not fully explore alternative solutions or adjustments to rectify these issues by extensively tuning the model architecture or designing more effective feature extraction process. Moreover, practical constraints, including limited computational resources, restricted the depth of hyperparameter optimization for both classification models—CNN-LSTM and RF+SVM ensemble, and these constraints forced us to use less number of epochs and a less intensive model architecture. While our simplified setups allowed us to complete the experiments effectively, this approach may have impacted the performance outcomes, potentially overlooking optimal model configurations[60].

Additionally, although we carefully chosen three prominent compression methods (lossless, wavelet-based, and autoencoder-based), we acknowledge this selection does not represent all available significant categories of ECG compression techniques, and we also did not achieve the same level of metrics claimed by the corresponding publications. Advanced adaptive compression methods or hybrid deep-learning approaches might yield additional insights, which were not covered in this study.

Lastly, while our analysis predominantly focused on obtaining all important metrics of the classification models, there are additional clinically meaningful criteria that this thesis did not address, such as model interpretability, robustness under real-time operational conditions, and user-specific adaptability. Another limitation of our study is the discrepancy in classification characteristics between the two baseline models: the RF-SVM ensemble model classifies five classes, whereas the CNN-LSTM model classifies only three classes. For a more accurate comparison, both models should have been designed to classify the same five heartbeat classes.These areas need to further investigation to enhance the practical clinical relevance of our results. Despite these acknowledged constraints, the outcomes of this thesis provide a valuable foundation for future exploration into ECG signal compression and its effective integration into classification models.

5.2 Future Directions

To precisely understand how compression affects classification models, future experiments should incorporate a more comprehensively implemented broad range of ECG compression techniques. This includes evaluating various combination-based compression methods, such as compressive sensing with dictionary-based approaches, to determine their optimal performance with specific classification model. Additionally, recent advancements in ECG compression techniques, particularly those suitable for real-time edge deployment, should be explored in future experiments. Future research could explore applying these state-of-the-art compression methods directly to the classification models, with subsequent deployment and evaluation on real-time edge devices. This approach would not only optimize model performance but also enhance practical applicability in clinical settings.

Furthermore, exploring additional deep-learning architectures, including transformer-based models, is necessary to make the current experimental setup genuinely beneficial. We highly recommend designing specific compression techniques for specific classification models and possibly integrating compression within the deep learning model architecture.

Chapter 5: Conclusion

In our current experiment, we have explored the MIT-BIH dataset across different compression techniques. However, this single dataset is insufficient to extensively examine the clinical impact of ECG compression on deep-learning-based classification models. For the future development of our study, it is highly recommended to examine the compression techniques with various other clinically significant databases such as the CPSC Database, CPSC-Extra Database, INCART Database, PTB and PTB-XL Databases. As these databases are quite different from the MIT-BIH database and many are not labeled beat-by-beat, further minor modifications to our experimental setup will be required. However, while designing our current experiment, we significantly emphasized reproducibility; therefore, only modifications to the data loading logic and preprocessing will be needed to incorporate these databases into our experiment. By incorporating more diverse and larger datasets, the validation and generalizability, along with the clinical applicability of the impact of compression on classification models, will be justified.

Conclusion

In this thesis, we systematically examined how various ECG signal compression techniques—lossless, wavelet-based, and autoencoder-based—impact the performance of the traditional RF+SVM ensemble arrhythmia classifier and the performance of the deep learning-based CNN-LSTM arrhythmia classifier. Our findings demonstrated that ECG compression does not affect the binary model size but can substantially influence classifier accuracy and indirectly slightly influence the inference efficiency of the deep learning-based CNN-LSTM arrhythmia classification model. Compression techniques in some cases may impact the model size and inference time of the traditional RF+SVM ensemble model due to the subtle changes in the features of the compressed ECG data although subsequent decompression after compression ensures the same shape and architecture of the compressed data. Compression-induced ECG signal might change the feature signatures in such a way that the changes impacted the RF+SVM ensemble model. We found traditional classifiers to be particularly sensitive to compressed test data when the model is trained on the original uncompressed signal. Conversely, our CNN-LSTM, deep learning model, particularly exhibited good results when the model was trained on the original ECG data and tested on the losslessly compressed ECG data. In all the other scenarios when the wavelet or autoencoder-based compressed signal was used in the deep learning-based CNN-LSTM model, the performance of the model was reduced. However, we found that the CNN-LSTM model is more vulnerable in terms of accuracy when aggressive compression techniques like autoencoder having PRD more than 7 percent are used. On the other hand RF+SVM ensemble model performed the best when the autoencoder compressed training data and test data was used. It is explicitly recommended from this study to strategically select compression methods with appropriate classification models, along with appropriate combination of compressed or uncompressed training and test data combination. Further robust study is required to comprehensively find which compression technique is best for specific machine learning models. These insights collectively contribute toward developing efficient, reliable, and clinically applicable AI-based ECG classification systems, underscoring the importance of carefully selecting compression methods aligned with classifier characteristics.

References

[1] A. Y. Hannun *et al.*, “Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network,” *Nat. Med.*, vol. 25, no. 1, pp. 65–69, Jan. 2019, doi: 10.1038/s41591-018-0268-3.

[2] M. Elgendi, A. Mohamed, and R. Ward, “Efficient ECG Compression and QRS Detection for E-Health Applications,” *Sci. Rep.*, vol. 7, no. 1, p. 459, Mar. 2017, doi: 10.1038/s41598-017-00540-x.

[3] C. K. Jha and M. H. Kolekar, “Tunable Q-wavelet based ECG data compression with validation using cardiac arrhythmia patterns,” *Biomed. Signal Process. Control*, vol. 66, p. 102464, Apr. 2021, doi: 10.1016/j.bspc.2021.102464.

[4] P. V. Reddy and V. V. S. Tallapragada, “FPGA-enabled lossless ECG signal compression system using an integer adaptive compressor,” *Analog Integr. Circuits Signal Process.*, vol. 119, no. 2, pp. 331–361, May 2024, doi: 10.1007/s10470-024-02269-w.

[5] L. E. Van Der Maarel and V. M. Christoffels, “Development of the Cardiac Conduction System,” in *Congenital Heart Diseases: The Broken Heart*, vol. 1441, S. Rickert-Sperling, R. G. Kelly, and N. Haas, Eds., in Advances in Experimental Medicine and Biology, vol. 1441. , Cham: Springer International Publishing, 2024, pp. 185–200. doi: 10.1007/978-3-031-44087-8_10.

[6] S. A. George *et al.*, “At the Atrioventricular Crossroads: Dual Pathway Electrophysiology in the Atrioventricular Node and its underlying Heterogeneities,” *Arrhythmia Electrophysiol. Rev.*, vol. 6, no. 4, p. 179, 2017, doi: 10.15420/aer.2017.30.1.

[7] N. Makris, Z. Kotsialou, and S. Gall, “Origin and regulation of the normal heartbeat,” *Anaesth. Intensive Care Med.*, vol. 25, no. 3, pp. 215–218, Mar. 2024, doi: 10.1016/j.mpaic.2023.11.015.

[8] M. Morad and X. Zhang, “Mechanisms of spontaneous pacing: sinoatrial nodal cells, neonatal cardiomyocytes, and human stem cell derived cardiomyocytes,” *Can. J. Physiol. Pharmacol.*, vol. 95, no. 10, pp. 1100–1107, Oct. 2017, doi: 10.1139/cjpp-2016-0743.

[9] M. Douard, “New insights into heart failure during this challenging time,” *J. Physiol.*, vol. 599, no. 13, pp. 3263–3265, Jul. 2021, doi: 10.1113/JP281300.

[10] G. Swapna, D. N. Ghista, R. J. Martis, A. P. C. Ang, and S. V. Sree, "ECG SIGNAL GENERATION AND HEART RATE VARIABILITY SIGNAL EXTRACTION: SIGNAL PROCESSING, FEATURES DETECTION, AND THEIR CORRELATION WITH CARDIAC DISEASES," *J. Mech. Med. Biol.*, vol. 12, no. 04, p. 1240012, Sep. 2012, doi: 10.1142/S021951941240012X.

[11] Servier Medical Art, "Diagram of the cardiac conduction system, including SA node, AV node, bundle branches, and Purkinje fibers [Illustration]," presented at the SMART-Heart-physiology,

[12] A. R. Houghton and D. Gray, "Electrocardiography," in *Oxford Textbook of Medicine*, 6th ed., J. Firth, C. Conlon, and T. Cox, Eds., Oxford University PressOxford, 2020. doi: 10.1093/med/9780198746690.003.0342.

[13] A. Abdou and S. Krishnan, "Horizons in Single-Lead ECG Analysis From Devices to Data," *Front. Signal Process.*, vol. 2, p. 866047, Apr. 2022, doi: 10.3389/frsip.2022.866047.

[14] M. Borloz, "Abnormalities of the P Wave and PR Interval," in *Electrocardiogram in Clinical Medicine*, 1st ed., W. J. Brady, M. J. Lipinski, A. E. Darby, M. C. Bond, N. P. Charlton, K. Hudson, and K. Williamson, Eds., Wiley, 2020, pp. 383–396. doi: 10.1002/9781118754511.ch37.

[15] G. Bagliani, R. De Ponti, C. Gianni, and L. Padeletti, "The QRS Complex," *Card. Electrophysiol. Clin.*, vol. 9, no. 3, pp. 453–460, Sep. 2017, doi: 10.1016/j.ccep.2017.05.005.

[16] M. G. Carey, "Acute Coronary Syndrome and ST Segment Monitoring," *Crit. Care Nurs. Clin. North Am.*, vol. 28, no. 3, pp. 347–355, Sep. 2016, doi: 10.1016/j.cnc.2016.04.006.

[17] L. H. Malik, G. D. Singh, and E. A. Amsterdam, "T-wave Tease: Apical Hypertrophic Cardiomyopathy," *Am. J. Med.*, vol. 127, no. 6, pp. 498–500, Jun. 2014, doi: 10.1016/j.amjmed.2014.02.028.

[18] F. Monitillo, "Ventricular repolarization measures for arrhythmic risk stratification," *World J. Cardiol.*, vol. 8, no. 1, p. 57, 2016, doi: 10.4330/wjc.v8.i1.57.

[19] C. Pater, "Methodological considerations in the design of trials for safety assessment of new drugs and chemical entities," *Curr. Control. Trials Cardiovasc. Med.*, vol. 6, no. 1, p. 1, Dec. 2005, doi: 10.1186/1468-6708-6-1.

[20] E. Al. Sandhya Samant, "Exploring ECG Signal Analysis Techniques for Arrhythmia Detection: A Review," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 11, no. 9,

pp. 4881–4896, Nov. 2023, doi: 10.17762/ijritcc.v11i9.10084.

[21] K. A. Mayuga *et al.*, “Sinus Tachycardia: a Multidisciplinary Expert Focused Review,” *Circ. Arrhythm. Electrophysiol.*, vol. 15, no. 9, Sep. 2022, doi: 10.1161/CIRCEP.121.007960.

[22] C. Kumar Jha and M. H. Kolekar, “Diagnostic quality assured ECG signal compression with selection of appropriate mother wavelet for minimal distortion,” *IET Sci. Meas. Technol.*, vol. 13, no. 4, pp. 500–508, Jun. 2019, doi: 10.1049/iet-smt.2018.5217.

[23] O. El B’charri, R. Latif, W. Jenkal, and A. Abenaou, “The ECG Signal Compression Using an Efficient Algorithm Based on the DWT,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 3, 2016, doi: 10.14569/IJACSA.2016.070325.

[24] M. M. Abo-Zahhad, T. K. Abdel-Hamid, and A. M. Mohamed, “Compression of ECG Signals Based on DWT and Exploiting the Correlation between ECG Signal Samples,” *Int. J. Commun. Netw. Syst. Sci.*, vol. 07, no. 01, pp. 53–70, 2014, doi: 10.4236/ijcns.2014.71007.

[25] T.-H. Tsai and W.-T. Kuo, “An Efficient ECG Lossless Compression System for Embedded Platforms With Telemedicine Applications,” *IEEE Access*, vol. 6, pp. 42207–42215, 2018, doi: 10.1109/ACCESS.2018.2858857.

[26] A. Vosoughi, M. Wu, and J. R. Cavallaro, “Baseband signal compression in wireless base stations,” in *2012 IEEE Global Communications Conference (GLOBECOM)*, Anaheim, CA, USA: IEEE, Dec. 2012, pp. 4505–4511. doi: 10.1109/GLOCOM.2012.6503828.

[27] C. Hernando-Ramiro, M. Blanco-Velasco, L. Lovisolo, and F. Cruz-Roldán, “Consistent quality control in ECG compression by means of direct metrics,” *Physiol. Meas.*, vol. 36, no. 9, pp. 1981–1994, Sep. 2015, doi: 10.1088/0967-3334/36/9/1981.

[28] J.-H. Hsieh, K.-C. Hung, J.-H. Liu, and T.-C. Wu, “Wavelet-Based Quality-Constrained ECG Data Compression System Without Decoding Process,” *IEEE Multimed.*, vol. 27, no. 2, pp. 33–45, Apr. 2020, doi: 10.1109/MMUL.2020.2983690.

[29] A. A. Pulavskyi, S. S. Krivenko, and L. S. Kryvenko, “Determination of the Signal-to-Noise Ratio for Noisy Electrocardiogram Using Lossless Data Compression,” in *2019 8th Mediterranean Conference on Embedded Computing (MECO)*, Budva, Montenegro: IEEE, Jun. 2019, pp. 1–4. doi: 10.1109/MECO.2019.8760294.

[30] V. Aggarwal, S. Gupta, M. S. Patterh, and Shipra, “Meta-Analysis approach to assess ECG compression methods: A systematic Review,” in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*,

Delhi, India: IEEE, Jul. 2023, pp. 1–10. doi: 10.1109/ICCCNT56998.2023.10307940.

[31] C. Crispin-Bailey, C. Dai, and J. Austin, “A 65-nm CMOS Lossless Bio-Signal Compression Circuit With 250 FemtoJoule Performance Per Bit,” *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 5, pp. 1087–1100, Oct. 2019, doi: 10.1109/TBCAS.2019.2938672.

[32] R. P. Tripathi and G. R. Mishra, “Study of various data compression techniques used in lossless compression of ECG signals,” in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, Greater Noida: IEEE, May 2017, pp. 1093–1097. doi: 10.1109/CCAA.2017.8229958.

[33] M. Jia, F. Li, Y. Pu, and Z. Chen, “A Lossless Electrocardiogram Compression System Based on Dual-Mode Prediction and Error Modeling,” *IEEE Access*, vol. 8, pp. 101153–101162, 2020, doi: 10.1109/ACCESS.2020.2998608.

[34] T.-H. Tsai and F.-L. Tsai, “Efficient lossless compression scheme for multi-channel ECG signal processing,” *Biomed. Signal Process. Control*, vol. 59, p. 101879, May 2020, doi: 10.1016/j.bspc.2020.101879.

[35] S. Banerjee and G. K. Singh, “A new real-time lossless data compression algorithm for ECG and PPG signals,” *Biomed. Signal Process. Control*, vol. 79, p. 104127, Jan. 2023, doi: 10.1016/j.bspc.2022.104127.

[36] K. Bannajak, N. Theera-Umpon, and S. Auephanwiriyakul, “Signal Acquisition-Independent Lossless Electrocardiogram Compression Using Adaptive Linear Prediction,” *Int. J. Environ. Res. Public. Health*, vol. 20, no. 3, p. 2753, Feb. 2023, doi: 10.3390/ijerph20032753.

[37] O. Yildirim, R. S. Tan, and U. R. Acharya, “An efficient compression of ECG signals using deep convolutional autoencoders,” *Cogn. Syst. Res.*, vol. 52, pp. 198–211, Dec. 2018, doi: 10.1016/j.cogsys.2018.07.004.

[38] F. Wang *et al.*, “A novel ECG signal compression method using spindle convolutional auto-encoder,” *Comput. Methods Programs Biomed.*, vol. 175, pp. 139–150, Jul. 2019, doi: 10.1016/j.cmpb.2019.03.019.

[39] J. Hua, J. Zou, J. Rao, H. Yin, and J. Chen, “ECG Signals Deep Compressive Sensing Framework Based on Multiscale Feature Fusion and SE Block,” *IEEE Access*, vol. 11, pp. 104359–104372, 2023, doi: 10.1109/ACCESS.2023.3316487.

[40] J. Shi *et al.*, “New ECG Compression Method for Portable ECG Monitoring System Merged with Binary Convolutional Auto-Encoder and Residual Error Compensation,” *Biosensors*, vol. 12, no. 7, p. 524, Jul. 2022, doi: 10.3390/bios12070524.

[41] C. Tan, L. Zhang, and H. Wu, “A Novel Blaschke Unwinding Adaptive-Fourier-Decomposition-Based Signal Compression Algorithm With Application on ECG Signals,” *IEEE J. Biomed. Health Inform.*, vol. 23, no. 2, pp. 672–682, Mar. 2019, doi: 10.1109/JBHI.2018.2817192.

[42] H. S. Pal, A. Kumar, A. Vishwakarma, G. K. Singh, and H.-N. Lee, “A new automated compression technique for 2D electrocardiogram signals using discrete wavelet transform,” *Eng. Appl. Artif. Intell.*, vol. 133, p. 108123, Jul. 2024, doi: 10.1016/j.engappai.2024.108123.

[43] H. S. Pal, A. Kumar, A. Vishwakarma, and M. K. Ahirwal, “Electrocardiogram signal compression using tunable-Q wavelet transform and meta-heuristic optimization techniques,” *Biomed. Signal Process. Control*, vol. 78, p. 103932, Sep. 2022, doi: 10.1016/j.bspc.2022.103932.

[44] M. H. Kolekar, C. K. Jha, and P. Kumar, “ECG Data Compression Using Modified Run Length Encoding of Wavelet Coefficients for Holter Monitoring,” *IRBM*, vol. 43, no. 5, pp. 325–332, Oct. 2022, doi: 10.1016/j.irbm.2021.10.001.

[45] R. K. Hamza, K. S. Rijab, and M. A. Hussien, “The ECG data Compression by Discrete Wavelet Transform and Huffman Encoding,” in *2021 7th International Conference on Contemporary Information Technology and Mathematics (ICCITM)*, Mosul, Iraq: IEEE, Aug. 2021, pp. 75–81. doi: 10.1109/ICCITM53167.2021.9677704.

[46] H. Zhang, Z. Dong, Z. Wang, L. Guo, and Z. Wang, “CSNet: A deep learning approach for ECG compressed sensing,” *Biomed. Signal Process. Control*, vol. 70, p. 103065, Sep. 2021, doi: 10.1016/j.bspc.2021.103065.

[47] T. R and V. B, “A novel ECG signal compression using wavelet and discrete anamorphic stretch transforms,” *Biomed. Signal Process. Control*, vol. 71, p. 102773, Jan. 2022, doi: 10.1016/j.bspc.2021.102773.

[48] R. Mahajan and D. Bansal, “Hybrid ECG signal compression system: A step towards efficient tele-cardiology,” in *2014 International Conference on Reliability Optimization and Information Technology (ICROIT)*, Faridabad, Haryana, India: IEEE, Feb. 2014, pp. 437–442. doi: 10.1109/ICROIT.2014.6798380.

[49] S. S. Kumar and P. Ramachandran, “Review on Compressive Sensing Algorithms for ECG Signal for IoT Based Deep Learning Framework,” *Appl. Sci.*, vol. 12, no. 16, p. 8368, Aug. 2022, doi: 10.3390/app12168368.

[50] L. Montenegro, M. Abreu, A. Fred, and J. M. Machado, “Human-Assisted vs. Deep Learning Feature Extraction: An Evaluation of ECG Features Extraction Methods for

Arrhythmia Classification Using Machine Learning," *Appl. Sci.*, vol. 12, no. 15, p. 7404, Jul. 2022, doi: 10.3390/app12157404.

[51] M. Sraitih, Y. Jabrane, and A. Atlas, "An overview on machine learning methods for ECG Heartbeat Arrhythmia Classification," in *2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA)*, Marrakech, Morocco: IEEE, Jun. 2021, pp. 8–14. doi: 10.1109/ICDATA52997.2021.00012.

[52] F. S. Butt, M. F. Wagner, J. Schafer, and D. G. Ullate, "Toward Automated Feature Extraction for Deep Learning Classification of Electrocardiogram Signals," *IEEE Access*, vol. 10, pp. 118601–118616, 2022, doi: 10.1109/ACCESS.2022.3220670.

[53] A. Eleyan and E. Alboghbaish, "Electrocardiogram Signals Classification Using Deep-Learning-Based Incorporated Convolutional Neural Network and Long Short-Term Memory Framework," *Computers*, vol. 13, no. 2, p. 55, Feb. 2024, doi: 10.3390/computers13020055.

[54] H. A. Shah, F. Saeed, M. Diyan, N. A. Almujally, and J. Kang, "ECG-TransCovNet: A hybrid transformer model for accurate arrhythmia detection using Electrocardiogram signals," *CAAI Trans. Intell. Technol.*, p. cit2.12293, Feb. 2024, doi: 10.1049/cit2.12293.

[55] *Multilabel classification*. New York, NY: Springer Berlin Heidelberg, 2016.

[56] F. S. Nahm, "Receiver operating characteristic curve: overview and practical use for clinicians," *Korean J. Anesthesiol.*, vol. 75, no. 1, pp. 25–36, Feb. 2022, doi: 10.4097/kja.21209.

[57] M. M. Farag, "A Self-Contained STFT CNN for ECG Classification and Arrhythmia Detection at the Edge," *IEEE Access*, vol. 10, pp. 94469–94486, 2022, doi: 10.1109/ACCESS.2022.3204703.

[58] A. Ukil *et al.*, "Resource Constrained CVD Classification Using Single Lead ECG On Wearable and Implantable Devices," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, Mexico: IEEE, Nov. 2021, pp. 886–889. doi: 10.1109/EMBC46164.2021.9630348.

[59] S. Alinsaif, "Unraveling Arrhythmias with Graph-Based Analysis: A Survey of the MIT-BIH Database," *Computation*, vol. 12, no. 2, p. 21, Jan. 2024, doi: 10.3390/computation12020021.

[60] Y. Cheng, Y. Ye, M. Hou, W. He, and T. Pan, "Multi-label Arrhythmia Classification from Fixed-length Compressed ECG Segments in Real-time Wearable ECG Monitoring," in *2020 42nd Annual International Conference of the IEEE Engineering in*

Medicine & Biology Society (EMBC), Montreal, QC, Canada: IEEE, Jul. 2020, pp. 580–583. doi: 10.1109/EMBC44109.2020.9176188.