

ECG pattern recognition based on polygon approximation techniques

Farah Isam Ali ^{1*}, Ekhlas K. Gbashi ¹, Ali Adel Saied ¹

¹College of Computer Science, University of Technology, Baghdad, Iraq

ARTICLE INFO

Received: 27/09/2025
Revised: 02/11/2025
Accepted: 27/11/2025
Available online: 08/08/2026
August Issue
[10.37652/juaps.2025.165575.1720](https://doi.org/10.37652/juaps.2025.165575.1720)

 CITE @ JUAPS

Corresponding author

Farah Isam Ali
cs.24.02@grad.uotechnology.edu.iq

ABSTRACT

The identification of patterns in electrocardiograms (ECGs) involves analyzing ECG signals through deep learning (DL) or machine learning (ML) methods to recognize specific patterns associated with various heart conditions. This study focuses on diagnosing heart disease patterns using polygon approximation techniques to extract features from ECG signals, offering a novel approach that hasn't been explored in the realm of ECG pattern recognition with machine learning classification methods. We evaluate two dominant point techniques. The first technique involves polygon approximation, and the second technique is based on curvature, implemented through the Ramer Douglas Peucker (RDP) algorithm and the Rosenfeld and Johnston (R&J) algorithm, respectively. This evaluation is conducted using two parameters: execution time and classification accuracy. We used several ML algorithms for the classification task, such as XGBoost, Random Forest (RF), Support Vector Machines (SVMs) with different kernels, and K-Nearest Neighbors (KNNs). RF classifier in conjunction with the Ramer Douglas Peucker approximation produced the most significant classification accuracy of 98.5%.

Keywords: Dominant points, ECG, Pattern recognition, Ramer Douglas Peucker, Rosenfeld and Johnston

1 INTRODUCTION

One of the most dangerous disorders worldwide is cardiovascular disease (CVD). Evidence from medicine has demonstrated that some medical disorders raise a person's risk of CVD, especially heart disease [1]. Because of this, clinical decision support systems (CDSS) can be very helpful in assisting doctors in making early diagnoses, and the digitization of healthcare systems is essential to preventing potentially deadly outcomes. [2,3]. Electrodes are placed at specific points on the body to record electrical signals generated by the heart, known as an electrocardiogram (ECG) [4,5]. An ECG typically has three primary waves: the P wave, the QRS complex, and the T wave [6]. As seen in Figure 1.

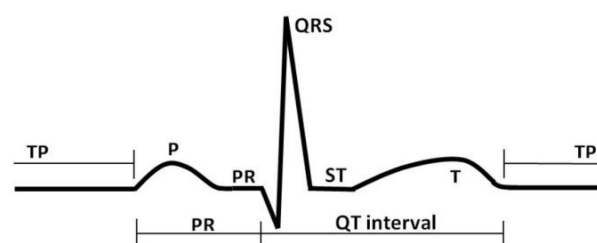


Fig. 1 A typical electrocardiogram signal.

Early detection of CVDs is greatly aided by pattern recognition as well as modeling in ECG signals [7]. For identifying heart diseases, numerous researchers have used artificial intelligence (AI) for analyzing ECG signals using DL, for example, the Convolution Neural Network (CNN) or ML algorithms [8–11].

To diagnose cardiac diseases, it is essential to read and analyze ECG signals, which show electrical patterns specific to each clinical condition. Usually, several steps are involved in identifying such patterns, including feature extraction and classification with ML or DL algorithms. Polygon approximation is a geometric simplification technique that represents complex curves or signals using fewer key points while preserving the overall shape and basic structural properties of the original data. In the context of digital signal processing, this technique reduces the number of points in a signal by replacing curved segments with straight segments that connect specific points of the original signal [12]. Despite the promise of this approach, the use of polygon approximation techniques in the context of ECG signals with machine learning algorithms has not yet been sufficiently explored. There were certain early attempts to target biomarkers. For example, Lee et al. (2018) used polygonal approximation methods to precisely and effectively locate fiducial points in the QRS complex of ECG signals. They implemented an algorithm for efficiently selecting fiducial points from ECG signals, incorporating polygonal approximation based on the curvature of the signal and employing dynamic programming to enhance the selection process [13]. It was still constrained, though, in terms of the variety of methods examined or classification algorithms applied. The objective of the present study is to investigate the efficacy of two major dominant point techniques for extracting features from ECG signals: the Rosenfeld and Johnston (R&J) algorithm and the Ramer Douglas Peucker (RDP) algorithm. For classifying four categories of cardiac conditions—Abnormal Heartbeat (HB), Myocardial Infarction (MI), Normal Heartbeat (Normal), and History of Myocardial Infarction (PMI)—feature vectors extracted with the use of these two techniques were assessed in conjunction with ML classifiers like XGBoost, RF, K-NN, and SVM. As far as we know, this is the first study to present a thorough scientific comparison of different ML classifiers and polygonal approximation-based feature extraction methods for ECG signals, accounting for both classification accuracy and execution time efficiency. The results have practical implications for real-time diagnostic applications and biomarker detection. Accordingly, the study is separated into the following sections: Related Work, Theoretical Background, Proposed Work, and Results and Discussion.

2 RELATED WORK

The following is a quick overview of related papers for ML classifiers used in ECG pattern recognition:

Aziz et al. extracted frequency-domain features from ECG signals using the Wavelet Transform (WT) and the Fast Fourier Transform (FFT); for enhanced signal analysis, they applied Two-Event-Related Moving Averages (TERMA) and the Fractional Fourier Transform (FrFT). For heartbeat classification, they employed the Multilayer Perceptron (MLP) and SVM, and the models were assessed utilizing the MIT-BIH and SPH databases. SVM achieved 99.6% accuracy on MIT-BIH and 37.1% on SPH using 36 features (AR+DWT), whereas MLP achieved 38.2% on SPH and 99.8% on MIT-BIH. SVM's accuracy decreased to 92.6% on MIT-BIH and 37% on SPH while using just four features (AR-based), while MLP achieved 93% on MIT-BIH and 39% on SPH [14]. For the classification of heart diseases based on ECG signals, a 2023 study compared several ML models, including Gaussian NB, LR, RF, Linear Discriminant Analysis (LDA), and a dummy classifier. With 96% accuracy, the overall results show that Gaussian NB outperformed all other ML models. Yet when characteristics are highly correlated, Gaussian NB might not perform well. Its poor feature-interaction modeling can lead to inaccurate predictions. Bertsimas et al. [15] developed a novel methodology to identify signal types in less than 30 ms by extracting features from ECG signals. Accurate predictions regarding heart disease were made using XGBoost ML models. The model's out-of-sample F1 scores range from 0.93 to 0.99. Dhyani et al. [16] employ the 3D DWT and SVM. The China Physiological Signal Challenge (CPSC) 2018 arrhythmia dataset is utilized for classifying ECGs using SVM for each of the 9 types of heartbeats that are identified by a variety of classifiers. With a normal precision of 99.02%, the SVM classifier outperformed the weighted support vector machine (WSVM) at 99% and the complex support vector machine (CSVM) at 98.5% [17]. CVD was predicted using six classifiers. Muniasamy and colleagues studied SVM, Artificial Neural Networks (ANN), k-NN, DTs, RF, and Linear Discriminant Analysis (LDA). Several datasets related to CVD were used to assess the classifiers. With the highest accuracy in predicting CVD, RF performed best [18]. To enhance the detection of myocardial infarction (MI), Wang et al. retrieved several features from ECG signals, including both temporal and frequency-domain features. The fused characteristics

were subjected to an RF classifier. To classify ECG signals and predict MI, our ML algorithm uses several DTs. Outperforming conventional techniques, the combination of RF and multi-feature fusion produced a high classification accuracy of 99.71% in identifying MI [19]. In this work, Hassaballah et al. utilized the MIT-BIH Arrhythmia Database. From ECG signals, several time- and frequency-domain characteristics have been retrieved. For distinguishing between different types of heartbeats, these qualities are essential. The parameters for four ML classifiers—SVM, k-NN, Gradient Boosting Decision Trees (GBDT), and RF—were optimized using metaheuristic algorithms, especially MHO-Metaheuristic Optimization. The modified model far outperformed conventional classification techniques, achieving an excellent average accuracy of 99.92% across many datasets [20]. The work by Subba et al. assessed ML algorithms for classifying ECG signals, with an emphasis on advanced feature extraction methods, such as the FFT, wavelet transform, and higher-order statistics. Signals have been divided into five primary categories using the MIT-BIH database. With 98% classification accuracy, the RF algorithm outperformed the SVM algorithm, which achieved 92% accuracy. Although they performed better than RF and SVM, DTs, logistic regression, and nearest neighbors all had respectable results [21]. A system for classifying ECG signals using waveform transformations and convolution with ML algorithms such as SVMs and ANNs was proposed in the paper by Marzog et al. The signals have been classified as congestive heart failure, arrhythmia, and normal sinus rhythm. With ANNs, the system's classification accuracy was 99.7%, while with SVMs, it was 99.92% [22]. The efficacy and robustness of ML algorithms in automatically identifying myocardial infarction in ECG data tainted by various forms of noise were the main areas of research for Sraitih et al. ECG data was analyzed using three well-known models: RF, KNN, and SVM. The PTB database's real ECG records have been utilized. The findings show that ML algorithms such as KNN, SVM, and RF are robust and reliable for identifying myocardial infarction from ECG data contaminated with noise [23]. To identify asymptomatic genetic disorders, Potter et al. developed an ML model

to analyze ECG signals. 643 characteristics have been recovered from the signals after they were converted into an "energy wave" (ewECG) format with the use of a CWT. The model was constructed using the ARIC study's heart failure risk score and 18 carefully selected characteristics via the RF algorithm. The model detected left ventricular dysfunction (LVD) with an AUC of 0.83, 72% specificity, and 85% sensitivity in the training set (287 subjects). [24] Gomes et al. investigated COVID-19 effects on heart function in the year 2023 by analyzing ECG signals using ML methods. Photographs of ECG signals from emergency cases were obtained from a public database. To classify patients as having myocardial infarction, a normal pulse, a history of infarction, COVID-19, or an aberrant pulse, random forests were employed after pre-trained CNNs were used to extract features. With a kappa index of 0.91, an AUC of 1.0, and 100% specificity and sensitivity, the model's classification accuracy was 94% [25]. With the use of ECG data, Kalmady et al. created AI models that predict 15 common cardiac diagnoses, including myocardial infarction, heart failure, atrial fibrillation, and ventricular tachycardia. Depending on ECG readings, a stepwise boosting (XGB) and a DL model (ResNet) were employed. With AUROC values above 90% in four diagnoses, the DL model outperformed the XGB model, which had AUROC values below 80% in the majority of diagnoses [26]. Using information such as gender, age, diabetes, and blood pressure, Hossain et al. used ML algorithms to identify ECG characteristics and predict heart disease. Variance, mean, and standard deviation have been extracted as statistical as well as morphological features. SVM, k-NN, logistic regression, Naïve Bayes, and an ensemble classifier (Ensemble Voting) were the five algorithms that were employed. When it came to predicting heart disease, SVM achieved the highest accuracy at 85.49%. [27] Rahma et al. classified five cases of heart disease: arrhythmia, ischemia, myocardial infarction, high blood cholesterol, and normal heart cases using UCI data and three supervised learning classification models: Naïve Bayes, SVM, and KNN. Based on the study's findings, NB was the optimal model, with a maximum accuracy of 96.9% [28]. Table 1 summarizes these studies.

Table 1 Summary of Reviewed ECG Analysis and Classification Studies

Study (Year)	Dataset(s)	Feature Extraction/ Domain	Algorithms Used	Accuracy
Aziz et al. (2023)	MIT-BIH, SPH	WT, FFT, TERMA, FrFT (Frequency-domain)	SVM, MLP	SVM: 99.6% (MIT-BIH), 37.1% (SPH); MLP: 99.8% (MIT-BIH), 38.2% (SPH)
Dhyani et al. (2023)	CPSC 2018	3D DWT	SVM, WSVM, CSVM	SVM: 99.02%, WSVM: 99%, CSVM: 98.5%
Subba et al. (2024)	MIT-BIH	FFT, WT, Higher-order statistics	RF, SVM, DT, LR, KNN	RF: 98%, SVM: 92%
Marzog et al. (2022)	MIT-BIH arrhythmia dataset	Waveform transformation, & convolution	SVM, ANN	SVM: 99.92%, ANN: 99.7%
Wang et al. (2020)	PTB MI ECG database	Temporal + Frequency-domain features	RF, DT	RF: 99.71% (MI detection)
Hassaballah et al. (2023)	MIT-BIH	Time + Frequency features	SVM, KNN, GBDT, RF (MHO-optimized)	Avg. accuracy: 99.92%
Hossain et al. (2021)	UCI repository datasets	Statistical & Morphological	SVM, KNN, LR, NB, Ensemble Voting	SVM: 85.49%
Rahma et al. (2022)	UCI dataset	Statistical	NB, SVM, KNN	NB: 96.9%
Muniasamy et al. (2020)	Multiple CVD datasets	- Clinical/statistical domain using cardiac patient data. - Numerical and structured features from patient records, without raw ECG signals.	SVM, ANN, KNN, DT, RF, LDA	RF best performer
Malakouti et al. (2023)	PTB-XL	Feature-domain/Morphological features extracted using pre-calculated ECG measurements.	Gaussian NB, LR, RF, LDA, Dummy	Gaussian NB: 96%
Bertsimas et al. (2021)	MIT-BIH	Novel ECG feature extraction	XGBoost	F1: 0.93–0.99
Kalmady et al. (2024)	Alberta Provincial ECG Database	- Time-domain (raw ECG signals) → for DL (ResNet). - Feature-domain (morphological/statistical ECG measurements) → for XGB.	XGB, ResNet (DL)	DL AUROC > 90%, XGB < 80%
Gomes et al. (2023)	Public emergency ECG images	CNN feature extraction (Image-based)	CNN + RF	Accuracy: 94%, AUC: 1.0, Kappa: 0.91
Potter et al. (2021)	ARIC study	CWT (ewECG format)	RF	AUC: 0.83, Sensitivity: 85%, Specificity: 72%
Sraitih et al. (2022)	PTB	Using real ECG signals from PTB after normalization and segmentation, with noise testing from the NSTDB database.	RF, SVM, KNN	Models robust against noisy ECG signals

3 MATERIALS AND METHODS

In our work, the following steps were followed for ECG pattern recognition, represented by a flow chart in Figure 2:

**Fig. 2** Proposed Work Steps of ECG Pattern Recognition

3.1 Ecg dataset

In the present study, we have utilized a 12-lead standard ECG dataset of cardiac patients from

Kaggle and Mendely, aiming to help the scientific community conduct research on CVDs [29–31]. This dataset represents four different subject classes belonging to the cardiac arrhythmia category:

1. ECG Images of Myocardial Infarction Patients (MI)
2. ECG Images of Patients with abnormal heart-beat (HB)
3. ECG Images of Patients with a History of Myocardial Infarction (PMI)
4. Normal Person ECG Images (Normal)

Data balance was achieved by undersampling the number of images of the largest class (Normal) from 1143 to 600 to match the smallest class (HB), which prevents the model from being biased towards the most numerous class and enhances its generalization ability. The number of PMI images and MI classes was also increased via augmentation, so that each class contains 600 images and all classes are equal after implementing the balance and optimization methods. Table 2 summarizes the image counts before and after undersampling and augmentation.

Table 2 Summary of the number of images before and after applying undersampling and augmentation procedures.

Class	No. of Available Images	No. of Images after undersampling	No. of Images after the augmentation	No. of Images Used in the Proposed System
HB	600	600	600	600
MI	277	277	600	600
PMI	284	284	600	600
Normal	1143	600	600	600

The database has been divided into training and test data with an 80% to 20% split. The total number of ECG images used for each class is listed in Table 3, and an example ECG Image is shown in Figure 3. To accurately analyze heart rhythms, a long recording from one of the leads, called a rhythm strip, is used. Lead II is typically chosen because it provides a clear representation of the P wave, which represents the activity of the atrial tubes, helping to assess the timing and consistency of contractions and accurately diagnose arrhythmias [32, 33].

Table 3 Summary of complexity and properties for the ViViT model's shapes

Classes	Training Data-set	Test Data-set	Total Data-set
MI	480	120	600
HB	480	120	600
PMI	480	120	600
Normal	480	120	600
Total	1,920	480	2,400

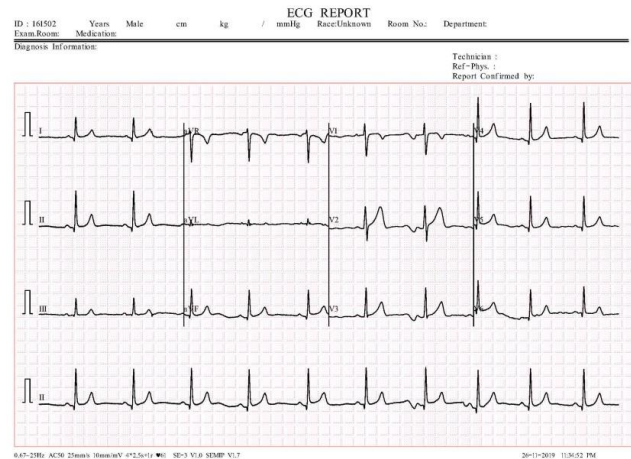


Fig. 3 One ECG Dataset Image

3.2 Preprocessing techniques

In this paper, we present the preprocessing steps performed on electrocardiogram (ECG) signal images to prepare them for analysis. These steps involve a set of digital image processing techniques implemented in Python, aiming to extract the signal shape cleanly and at a single-pixel width (skeleton).

- **Reading ECG Image:** Images were loaded from a dedicated folder containing ECG signal images saved in .jpg, .png, or .jpeg formats.
- **Cropping the lead II strip from the ECG signal image:** A rectangle containing the lead II strip is cropped using predefined coordinates.
- **Convert the ECG Image to a Grayscale Image:** The cropped image was converted to grayscale to reduce dimensionality, then to binary using a threshold to separate the signal from the background. The final grayscale value is calculated according to the following equation [34].

$$Y = 0.299 \times R + 0.587 \times G + 0.114 \times B$$

- **Binarization:** Convert the grayscale ECG Image to a binary image using a threshold technique. The grayscale image $I(x, y)$ can be converted to a binary image $B(x, y)$ utilizing

a specific threshold value T . The mathematical equation for this technique is defined as follows [35,36].

$$B(x, y) = \{255, I(x, y) \geq T \quad 0, I(x, y) < T$$

- Background removal: Morphological closing is used to remove background noise by reducing small gaps within the signal.
- Color inversion and skeletonization: The colors were reversed to match the requirements of the slimming algorithm, which expects the signal to be white and the background to be black, and then Lee's algorithm was applied to slim the signal using the skeletonize function to transform the signal into a centerline with a width of one pixel. The resulting image typically produces a simplified shape known as a skeleton, which retains the shape's path and center but consists of only a few pixels [37]. Figure 4 explains the steps of preprocessing:

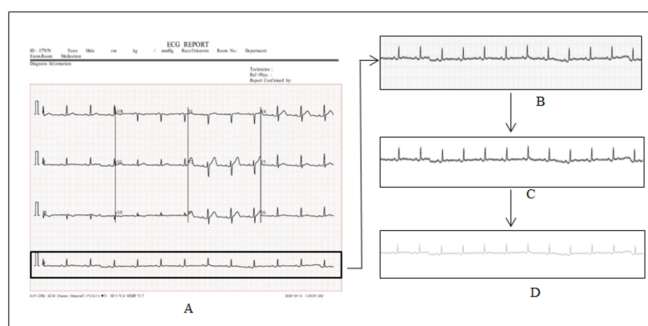


Fig. 4 reprocessing steps (A. ECG Signal Image, B. Crop and Convert to grayscale Image, C. Binarization, D. Skeletonization)

3.3 Feature extraction

In our work, we relied on extracting a set of approximation points to construct the feature vector from ECG images, based on classification labels: normal, abnormal, myocardial, and history of myocardial, using machine learning algorithms. With the first algorithm, the Ramer Douglas Peucker

algorithm, we pre-defined the tolerance value sequentially, starting from an initial value and gradually increasing it by 0.1 in each cooling until the required tolerance conditions were met. In other words, the Ramer-Douglas-Peucker algorithm was called repeatedly until we obtained the target number of approximation points (a set of points). In the second approach, we relied on constructing the feature vector using the Rosenfeld & Johnston algorithm, which is called only once, according to the algorithm's conditions, as explained below.

3.4 Polygon approximation

To obtain the properties of an object, firstly, its dominant points must be determined. Dominant points are high-curvature points on digital curves that capture their essential properties. These points are used to simplify the boundary representation and convert it to a polygon, facilitating image analysis operations such as recognition and matching [38]. The detection of dominant points is classified into two main categories: polygonal approximation methods and corner detection methods [39].

In polygonal approximation methods, dominant points are extracted after the approximation, representing the vertices of the asymptotic polygon. Polygonal approximation methods can be categorized into three main categories: the sequential approach, the split-and-merge approach, and the heuristic search approach. [40] Polygonal approximation can be classified into 2 classes of sub-problems [41]:

- Min-# problem: considering an n -vertex polygonal curve C , it is approximated with the use of another polygonal curve A with a minimal number of segments K , by considering that the approximation error E doesn't exceed a specific maximal value of tolerance E_0 .
- Min- ϵ problem: considering an n -vertex polygonal curve C , this curve is approximated with the use of another polygonal curve A with a specific number of line segments K , so

that the approximation error E is reduced to a minimum.

In corner detection methods, dominant points can be identified by measuring high curvature values using curvature-based metrics or alternative metrics, such as the k -cosine metric [40].

3.4.1 Ramer douglas peucker (rdp)

The Douglas-Peucker algorithm simplifies curves by reducing the number of points that represent the shape while preserving its overall shape [42, 43]. Ramer iteratively splits the curve into two smaller curves at the point with a maximal deviation to a point where the maximum deviation is less than a preset threshold [44]. The algorithm begins by marking the first and last points on the curve as "points of interest" (V_1 and V_n), then draws a straight line between them. It then finds the point between V_2 and V_{n-1} that has the greatest vertical distance from this line, called V_k . If this distance (error or deviation) at V_k is greater than or equal to a preset threshold ($\epsilon > 0$), then V_k is also considered significant. The algorithm then divides the curve into two parts: from V_1 to V_k and from V_k to V_n , applying the same steps repeatedly (iteratively) to each part. This process continues until no points deviate by more than the threshold, or until the part shrinks to a straight line between only 2 points. In this way, the algorithm produces a simplified version of the curve containing only the "necessary" points that maintain the overall shape [43, 45]. Below are the steps of the Douglas Peucker algorithm [46].

3.4.2 Rosenfeld and johnston polygon approximation

Rosenfeld and Johnston (1973) detect dominant points on digital curves by computing the cosine of the inner angle between forward and backward vectors of increasing support length. By selecting the smoothing parameter k , local maxima in that cosine sequence are identified as angles (sharp features) [47]. The Rosenfeld and Johnston method is a classical geometric technique used to estimate curvature at discrete points along a curve, particularly in digital image processing and shape analysis. This method

analyzes the inner angle at a point on the curve by computing the cosine of the angle between 2 vectors extending to the neighboring points. These vectors are formed by connecting the target point P_i to a point P_{i-k} to its left and a point P_{i+k} to its right, where k is a variable support distance [43], following the steps of the Rosenfeld and Johnston algorithm [48].

Algorithm 1: Douglas-Peucker algorithm

Input: Point List: A series of points representing the curve (e.g., $[P_1, P_2, \dots, P_n]$).

Threshold (ϵ): A value representing the maximum allowable error

Output: Simplified Curve: A list of points representing the simplified curve

begin

Draw a baseline: Connect the first point, P_1 , to the last point, P_n , with a straight line.;

Find the maximum deviation: Calculate the vertical distance from each point between P_1 and P_n to the baseline.;

Determine the point P_{max} with the largest distance, d_{max} , from the line.;

Check the threshold (ϵ): If $d_{max} > \epsilon$:
Keep point P_{max} .

Else divide the curve into two parts: from P_1 to P_{max} and from P_{max} to P_n ;

Apply the algorithm recursively to each part. If $d_{max} \leq \epsilon$: Delete all points between P_1 and P_n , keeping only the final two points (P_1 and P_n). ;

Combine the results: Combine the "retained" points from each iterative call to form the final simplified curve.;

All of the above aim to compare the first algorithm with the second algorithm to determine which provides better results and produces higher classification accuracy and lower time. In the recognition process, feature vectors are generated for both the training and test sets, as depicted in Figure 5. Every feature vector comprises the Cartesian coordinates of the approximation points representing each class.

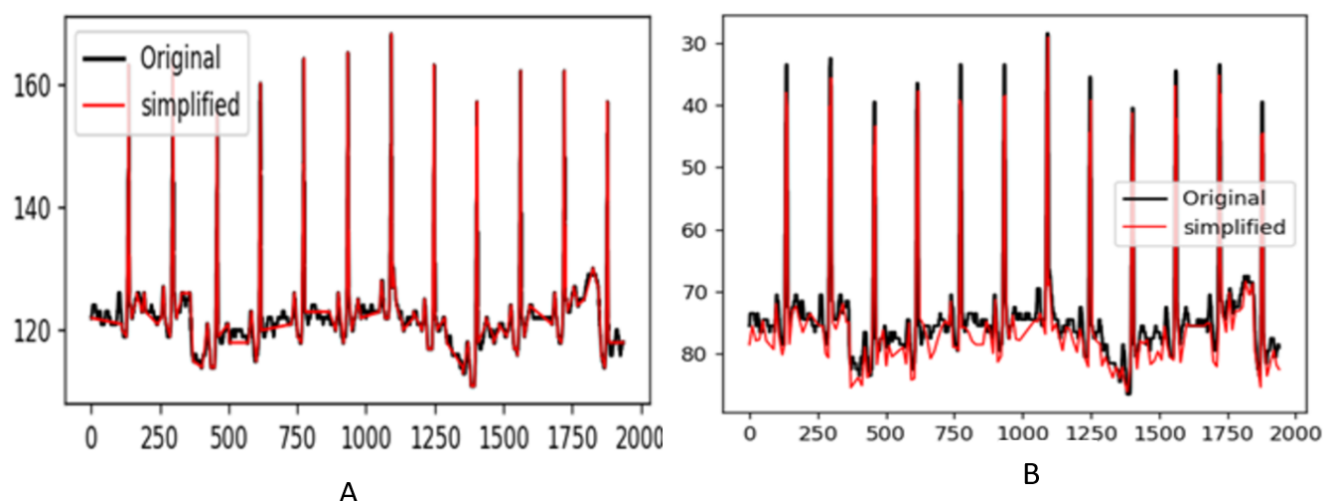


Fig. 5 Simplified feature vector by (A) Ramer-Douglas-Peucker (RDP) and (B) Rosenfeld and Johnston algorithms

3.5 Machine learning classifiers

In our work, we used four types of ML classifiers: KNN, SVM, Random Forest, and XGBoost, which are explained below.

3.5.1 K nearest neighbor (knn)

The KNN algorithm depends on a neighborhood of close (i.e., similar) patterns relative to the query pattern, and a significant challenge is finding the optimal measure of similarity or distance [49, 50]. The KNN (k-nearest neighbors) algorithm is implemented by calculating k-nearest neighbors (i.e., points) of a given sample, based on the shortest distance between them and this sample (query point). One technique to determine the value of k is to run the classification multiple times with different values to see which produces the best outcomes [51]. Four distinct distance metrics are employed with k values, namely, Euclidean, Minkowski, Cosine, and Chebyshev distance. The four distance metrics used for star classification are discussed in the following section [52].

- Euclidean Distance: $d(x, y) = \sqrt{2_{(ix-y)} \sum_{i=1}^n}$
- Minkowski Distance

$$d(x, y) = |iy - ix| \sum_{i=1}^n$$

- Cosine Distance $d(x, y) = \frac{x \cdot y}{\|x\| \cdot \|y\|} - 1$

- Chebyshev Distance $d_{\text{Chebyshev}} = (x, y) |iy - ix|$

3.5.2 Support vector machine (svm)

An SVM is a supervised learning method utilized in order to solve classification or regression tasks [53]. The SVM algorithm is based on three main principles of analysis: dimension separation in the feature space, projection calculation through inner multiplication using kernel functions, and optimization of the separation process to obtain support vectors. This method determines support vectors by constructing optimal boundaries in the feature space using a kernel function that minimizes errors across dimensions. Kernel functions are used to reduce feature dimensionality via two eigenvectors, such as linear functions, polynomials, or Gaussian radial basis functions (RBFs), to determine similarity across dimensions [54]. In this work, we used three kernel functions (K):

Algorithm 2: Rosenfeld and Johnston algorithm**begin**

Input Curve Let the digital curve be a sequence of 2D points:

$C = P_1, P_2, \dots, P_n$, where each

$P_i = (x_i, y_i)$;

Define a Smoothing Factor. Choose a smoothing parameter

$k \in (0, 1)$ (typical value : 0.05).

Compute maximum support length:

$m = k.n$;

For Each Point P_i , Compute Curvature

For $k = 1$ to m : Define vectors:

$$a^{-ik} = p_{i+k} - p_i, b^{-ik} = p_{i-k} - p_i \mid$$

Compute the cosine of the inner angle:

$$\cos_{ik} = \frac{a^{-ik} \cdot b^{-ik}}{\|a^{-ik}\| \cdot \|b^{-ik}\|}$$

$\cos_{ik} = +1$: sharpest angle (0°)

$\cos_{ik} = -1$: straight line (180°)

$\cos_{ik} \in [-1, 1]$

;

Determine Region of Support h_i For

every P_i , find the largest support

$h_i \in [1, m]$ such that:

$$\cos_{i1} < \cos_{i2} < \dots < \cos_{i(h_i-1)} \geq \cos_{ih_i}$$

This means the curvature (cosine) increases, peaks at h_i , then decreases. ;

Mark Candidate Dominant Points For

each point $p_{i'}$, keep the pair:

$(p_{i'}, \cos_{ih_i})$;

Non-Maximum Suppression Retain only

those points p_i where: $\cos_{ih_i} \geq \cos_{jh_j}$

For every j such that $|i - j| \leq h_i$ This

ensures that only the local maxima in curvature are selected as dominant

points.;

Output The final set of dominant points on the curve.

• Linear: $K(x, y) = x.y$

• Polynomial: $K(x, y) = (ax.y + b)$

• Radial Basis Function (RBF):

$$K(x, y) = \exp(-\|x - y\|^2 / \sigma^2)$$

3.5.3 Extreme gradient boosting (xgboost)

One of the most effective supervised learning algorithms is XGBoost. Through minimizing a specific loss function, it constructs a model that can link outputs and inputs [55]. This algorithm is designed to improve the performance of weak learners. XGBoost uses additive modeling, which is incremental and phase-wise. After fitting the data to an initial weak classifier, the model trains a new classifier to correct the errors caused by the first model without compromising the performance of the current model. Until the system reaches the highest level of accuracy, this procedure is repeated across several phases, adding each new model incrementally to the previous one [56, 57].

3.5.4 Random forest

It is an ensemble-based classification algorithm that consists of a large number of decision trees and extracts classification results from each tree [58]. The RF model constructs multiple trees rather than a single tree, with each tree generated from random vectors independently selected from the overall data distribution. This approach is known as stochastic learning, as training samples and input features are randomly selected for each tree, reducing overlap between submodels and increasing their diversity. In the prediction phase, the outputs of all the trees are combined—either by majority voting for classification or by averaging for regression—to produce the model's final decision [59, 60].

The classification step involves 2 steps. Training for generating learning features by assigning a distinguished label to each class. The classification of a test feature vector. The algorithm for Training and testing using the Ramer-Douglas-Peucker (RDP) algorithm is as follows:

Algorithm 3: Training and testing the model using the RDP algorithm

begin

Train: Repeat
 Read the ECG image from the input folder or list;
 Crop the long Lead II strip from the ECG image;
 Convert the image to grayscale;
 Convert the image to binary by utilizing a thresholding technique.;
 Apply morphological operations to remove the background;
 Apply skeletonization to the ECG line;
 Extract the (x, y) coordinates of the ECG signal ;
 Set tolerance value ϵ and initial value $=0$; with maximal value M .;
 While $(\epsilon < M)$ Do;
 Increase tolerance value $\epsilon = \epsilon + 0.01$, where $0 \leq \epsilon \leq 1$.;
 Compute dominant points with the use of the (RDP) algorithm;
 If $(M > \epsilon)$ then exit; else return to Step 11. Generate the feature vector (e.g., by concatenating x and y into a 1D vector);
 Store the feature vector for this image
 Until all ECG images have been processed;

begin

Test:
 Store the feature vector for the input ECG image;
 Load the training dataset (e.g., normal, abnormal, etc.);
 For each feature vector extracted from the ECG images

- Classify using one of the algorithms: KNN, SVM, Random Forest, or XGBoos.
- Save or return the class label of the beat

 Return all predicted classes.

The algorithm for training and testing the model with the use of the Rosenfeld and Johnston algorithm is as follows:

Algorithm 4: Training and testing the model using the RDP algorithm

begin

Train: Repeat
 Repeat Read the ECG image from the input folder or list;
 ;
 Crop the long Lead II strip from the ECG image;
 Convert the image into a greyscale;
 Convert the image into binary using a threshold;
 Apply morphological operations to remove the background;
 Extract contours of shape.;
 Simplify contour points ;
 Extract the (x, y) coordinates of the ECG signal;
 Generate the feature vector by R&J;
 Store the feature vector ;
 Until all ECG images have been processed;

begin

Test:
 Store the feature vector for the input ECG image Load the training dataset (e.g., normal, abnormal, etc.) For each feature vector extracted from the ECG images:

- Classify using one of the algorithms: KNN, SVM, Random Forest, or XGBoost
- Save or return the class label of the beat

 Return all predicted classes

4 RESULTS AND DISCUSSION

In this section, we present the accuracy results in tabular form. The accuracy obtained with the Ramer-Douglas-Peucker algorithm is shown in Tables 4, 5, and 6, while the results using the Rosenfeld & Johnston algorithm are shown in Tables 7, 8, and 9. The execution time results are illustrated in Figure 6 as follows:

Table 4 Accuracy of the Douglas-Peucker technique and KNN classifier

Polygon Technique	Number of Points	KNN Minkowski	KNN Euclidian	KNN chebyshev	KNN Cosine
Douglas-Peucker algorithm	100	95.83%	89.58%	25.00%	50.42%
	150	94.38%	90.42%	25.00%	53.33%
	200	93.96%	88.96%	25.00%	58.96%
	250	93.75%	89.17%	25.00%	70.21%

Table 4 shows the results of the KNN classifier using four types of distances (Minkowski, Euclidean, Chebyshev, and Cosine) after applying the Douglas Peucker algorithm at different levels of the number of points (100, 150, 200, 250). The Minkowski distance showed the highest accuracy (95.83%) at 100 points. With the Euclidean distance, the high accuracy achieved was 90.42% at 150 points. Chebyshev achieved low accuracy (25%) across all cases, demonstrating its weakness in this context. In contrast, Cosine gradually improved with increasing points, from 50.42% to 70.21%, reflecting its sensitivity to detail. In general, Minkowski and Cosine are best suited for this type of data.

Table 5 Accuracy of the Douglas-Peucker technique and SVM classifier

Polygon Techniques	Number of Points	SVM RPF	SVM Linear	SVM Polynomial
Douglas-Peucker algorithm	100	88.96%	76.25%	71.46%
	150	88.54%	76.25%	77.50%
	200	88.54%	84.58%	77.08%
	250	89.79%	91.88%	73.33%

Table 5 shows the results of shape classification using the SVM classifier with three kernel types: Polynomial, RBF, and Linear, after applying the Douglas Peucker algorithm at various levels of the

number of points (100, 150, 200, and 250). The results show that the accuracy of SVM RPF ranged from 88.5% to 89.7%, with the best performance at 250 points (89.79%). In contrast, the accuracy of SVM Linear improved significantly with increasing numbers of points, reaching a maximum of 91.88% at 250 points. The SVM Polynomial achieved accuracies between 71.46% and 77.50%, with the best result at 150 points.

Table 6 Accuracy of the Douglas-Peucker technique and Random Forest and XGBoost classifiers

Polygon Technique	Number of Points	Random Forest	XGBoost
Douglas-Peucker algorithm	100	98.54%	96.25%
	150	94.38%	94.79%
	200	92.92%	94.58%
	250	95.00%	94.38%

Table 6 displays the results of applying the Random Forest and XGBoost classifiers after applying the Douglas-Peucker algorithm at various point counts (100, 150, 200, and 250). Random Forest showed very high accuracy at 100 points (98.54%), but dropped to 92.92% at 200 points. In contrast, XGBoost showed accuracy ranging from 94.38% to 96.25% across all levels of points. Overall, both classifiers are effective.

Table 7 Accuracy of the Rosenfeld & Johnston technique and KNN classifier

Polygon Technique	Number of Points	KNN Minkowski	KNN Euclidian	KNN chebyshev	KNN Cosine
Rosenfeld & Johnston algorithm	100	94.38%	94.38%	93.54%	93.75%
	150	93.75%	93.54%	92.71%	93.33%
	200	93.54%	93.54%	94.58%	92.92%
	250	94.58%	93.33%	92.50%	94.58%

Table 7 presents the performance of the KNN classifier using four distance metrics (Minkowski, Euclidean, Chebyshev, and Cosine) when applying the Rosenfeld and Johnston algorithm. The Minkowski and Euclidean distances yielded very similar results, with classification accuracies ranging from 93% to 94%, while the Chebyshev distance showed lower accuracy, especially at 150 points (92.71%). The Cosine distance achieved the best result at 250 points (94.58%). These results indicate that the Rosenfeld

and Johnston technique preserves shape information well regardless of the distance metric used in KNN.

Table 8 Accuracy of the Rosenfeld & Johnston technique and SVM classifier

Polygon Technique	Number of Points	SVM RPF	SVM Linear	SVM Polynomial
Rosenfeld & Johnston algorithm	100	72.08%	69.58%	48.96%
	150	77.29%	71.04%	47.71%
	200	77.29%	76.25%	48.54%
	250	78.96%	78.75%	49.79%

Table 8 shows the results of using the SVM classifier with different kernel types (Polynomial, RBF, and Linear) and the Rosenfeld-Johnston algorithm with varying numbers of points. The results show that the accuracy of SVM RPF gradually improved from 72.08% at 100 points to 78.96% at 250 points, whereas the accuracy of SVM Linear improved from 69.58% to 78.75% as the number of points increased. However, the accuracy of the SVM Polynomial remained relatively low, ranging from 47.71% to 49.79%, without significant improvement, suggesting that increasing the number of points had a greater positive impact on RPF and linear classifiers than on the polynomial classifier.

Table 9 Accuracy of the Rosenfeld & Johnston technique and Random Forest and XGBoost classifiers

Polygon Technique	Number of Points	Random Forest	XGBoost
the Rosenfeld & Johnston algorithm	100	95.42%	95.42%
	150	95.62%	95.62%
	200	96.25%	95.21%
	250	96.25%	94.79%

Table 9 displays the results of the Random Forest

and XGBoost classifiers when using the Rosenfeld and Johnston algorithm at different point levels. With Random Forest, the accuracy was best at 200 and 250 points (96.25%), while XGBoost showed similar results, with 94.79% at 250 points. These results indicate that both classifiers effectively handle the shape representations generated by these algorithms.

In summary, the results showed that the Douglas-Peucker technique outperformed the Rosenfeld and Johnston technique in classification accuracy, achieving 98.54% with the Random Forest classifier at 100 points. The reason why the Random Forest classifier outperforms its peers with the Ramer Douglas Peucker algorithm is due to the feature vector that is built by the dominant point, depending on the tolerance that increases gradually by 0.1. Therefore, we did not rely on a single diversity measure, but rather on the best one that achieved the required number of points, which improved the approximation accuracy. Overall, the Douglas-Peucker technique is the most effective, especially with classifiers based on comprehensive data analysis, such as Random Forest and XGBoost.

Figure 6 shows the training time for two dominant point techniques for ECG feature extraction: Douglas-Peucker and Rosenfeld and Johnston. The training time was measured across four clinical categories (normal, abnormal, MI, and history of myocardial) and at different point levels (100, 150, 200, and 250 approximate points) using Python's time library. All experiments were run on a laptop equipped with an Intel Core i7-8565U CPU @ 1.80 GHz, 16 GB of RAM (2667 MHz), a 2 GB graphics card, and 238 GB of internal storage. The system ran Windows 10 Super Lite 64-bit.

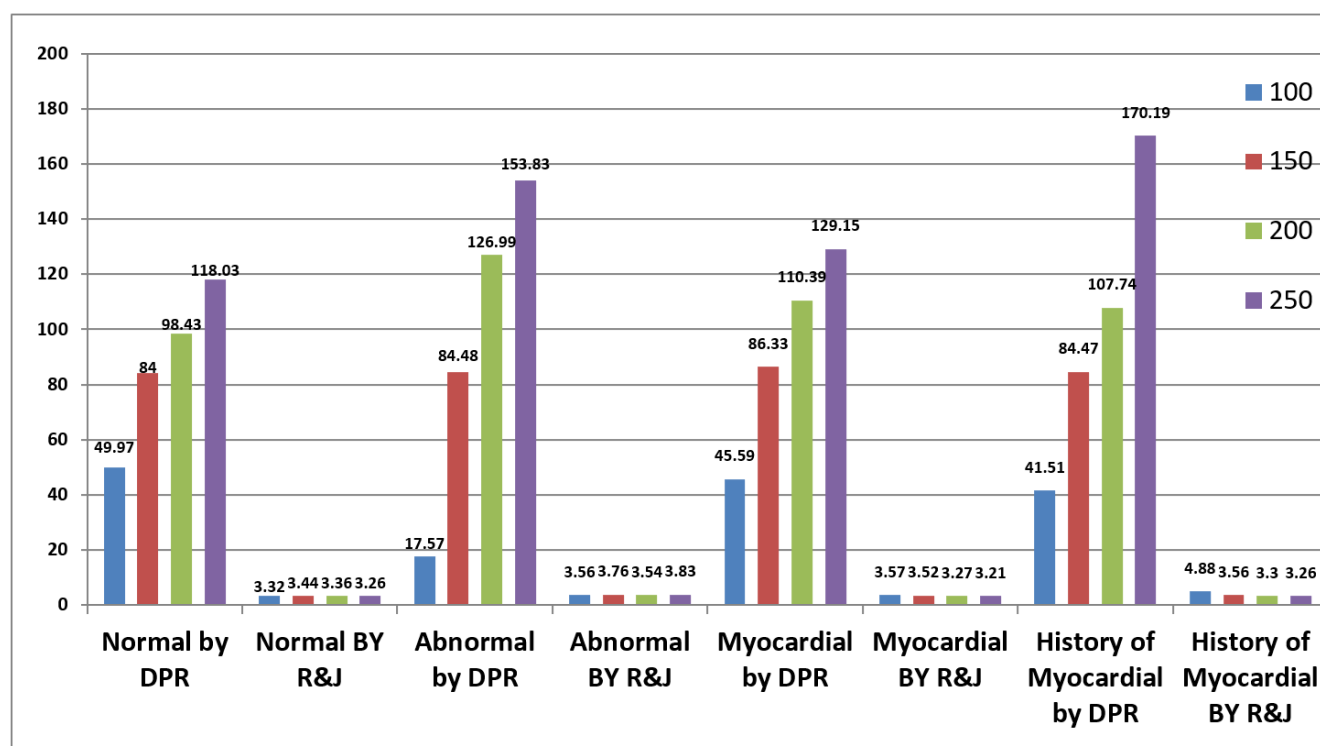


Fig. 6 Histogram of training time results with both polygon techniques in minutes.

The results indicated that the Douglas-Peucker approach necessitated considerably longer training durations than the Rosenfeld & Johnston algorithm, surpassing 170 minutes in certain categories, whilst the Rosenfeld & Johnston algorithm consistently remained under five minutes. It was also shown that using Douglas-Peucker with more points resulted in much longer training time, whereas using Rosenfeld & Johnston kept training time almost the same, indicating low time complexity and suitability for applications that need to be done quickly.

5 CONCLUSION

The proposed work presents a comprehensive examination of the training duration and classification accuracy of two strategies for simplifying curves generated from ECG signals: the Rosenfeld & Johnston algorithm and the Douglas-Peucker algorithm. The RDP method achieved the best classification accuracy of 98.54% with the RF classifier, but it took a long time to train, typically exceeding 170 minutes. Rosenfeld & Johnston's method, on the other hand,

was not very accurate but did save time (typically no more than 5 minutes). In this way, the Rosenfeld & Johnston method is better for applications that require fast processing or quick reaction, whereas the Douglas-Peucker method is better for achieving the most accurate results.

Author contributions

Ali Adel and Ekhlas Khalaf designed the study; Farah Isam conducted experiments, analyzed the data and wrote the manuscript.

Funding source

No funds received.

Data availability

N/A

DECLARATIONS

Conflict of interest

The authors declare no competing interests

Consent to publish

N/A

Ethical approval

N/A

REFERENCES

- [1] Azmi J, Arif M, Nafis MT, Alam MA, Tanweer S, Wang G. A Systematic Review on Machine Learning Approaches for Cardiovascular Disease Prediction Using Medical Big Data. *Medical Engineering & Physics*. 2022;105:103825. [10.1016/j.medengphy.2022.103825](https://doi.org/10.1016/j.medengphy.2022.103825)
- [2] Shobanadevi A, Veeramakali T. Classification and Interpretation of ECG Arrhythmia through Deep Learning Techniques; 2023. Version 1; preprint. Research Square preprint. [10.21203/rs.3.rs-2935678/v1](https://doi.org/10.21203/rs.3.rs-2935678/v1)
- [3] Addison T, Bhadrashetty A. Advanced Predictive Model for Heart Disease in Clinical Decision Support Systems. *EDRAAK*. 2023;2023:11-5. [10.70470/EDRAAK/2023/003](https://doi.org/10.70470/EDRAAK/2023/003)
- [4] Hampton J, Hampton J, Adlam D. *The ECG Made Easy*. 10th ed. Elsevier Health Sciences; 2024
- [5] Melzi P, Tolosana R, Vera-Rodriguez R. ECG Biometric Recognition: Review, System Proposal, and Benchmark Evaluation. *IEEE Access*. 2023;11:15555-66. [10.1109/ACCESS.2023.3244651](https://doi.org/10.1109/ACCESS.2023.3244651)
- [6] Wheed SF, Hayawi MJ. Classification of ECG Signal Using Deep Convolutional Neural. *Journal of Education for Pure Science–University of Thi-Qar*. 2022;12(2):64-72. [10.32792/jeps.v12i2.190](https://doi.org/10.32792/jeps.v12i2.190)
- [7] Muthalagu R, Ramachandran R, Anupama PH. Pattern Recognition and Modelling in Electrocardiogram Signals: Early Detection of Myocardial Ischemia and Infarction. In: 2023 2nd International Conference on Edge Computing and Applications (ICE-CAA). IEEE; 2023. p. 1035-41. [10.1109/ICE-CAA58104.2023.10212133](https://doi.org/10.1109/ICE-CAA58104.2023.10212133)
- [8] Anbalagan T, Nath MK, Vijayalakshmi D, Anbalagan A. Analysis of Various Techniques for ECG Signal in Healthcare, Past, Present, and Future. *Biomedical Engineering Advances*. 2023;6:100089. [10.1016/j.bea.2023.100089](https://doi.org/10.1016/j.bea.2023.100089)
- [9] Akbilgic O, Butler L, Karabayir I, Chang PP, Kitzman DW, Alonso A, et al. ECG-AI: Electrocardiographic Artificial Intelligence Model for Prediction of Heart Failure. *European Heart Journal – Digital Health*. 2021;2(4):626-34. [10.1093/ehjdh/ztab080](https://doi.org/10.1093/ehjdh/ztab080)
- [10] Khorsheed MS, Karim AA. Analyzing the Impact of Transfer Learning on Explanation Accuracy in Deep Learning-Based ECG Recognition Systems. *Open Engineering*. 2024;14(1):20240066. [10.1515/eng-2024-0066](https://doi.org/10.1515/eng-2024-0066)
- [11] Al Barazanchi II, Hashim W, Thabit R, Hussein NAHK. Advanced Hybrid Mask Convolutional Neural Network with Back-propagation Optimization for Precise Sensor Node Classification in Wireless Body Area Networks. *KHWARIZMIA*. 2024;2024:17-31. [10.70470/KHWARIZMIA/2024/004](https://doi.org/10.70470/KHWARIZMIA/2024/004)
- [12] Salotti JM. An Efficient Algorithm for the Optimal Polygonal Approximation of Digitized Curves. *Pattern Recognition Letters*. 2001;22(2):215-21. [10.1016/S0167-8655\(00\)00088-X](https://doi.org/10.1016/S0167-8655(00)00088-X)
- [13] Lee S, Jeong Y, Park D, Yun BJ, Park KH. Efficient Fiducial Point Detection of ECG QRS Complex Based on Polygonal Approximation. *Sensors*. 2018;18(12):4502. [10.3390/s18124502](https://doi.org/10.3390/s18124502)
- [14] Aziz S, Ahmed S, Alouini MS. ECG-Based Machine-Learning Algorithms for Heartbeat Classification. *Scientific Reports*. 2021;11(1):18738. [10.1038/s41598-021-97118-5](https://doi.org/10.1038/s41598-021-97118-5)

- [15] Malakouti SM. Heart Disease Classification Based on ECG Using Machine Learning Models. *Biomedical Signal Processing and Control*. 2023;84:104796. [10.1016/j.bspc.2023.104796](https://doi.org/10.1016/j.bspc.2023.104796)
- [16] Bertsimas D, Mingardi L, Stellato B. Machine Learning for Real-Time Heart Disease Prediction. *IEEE Journal of Biomedical and Health Informatics*. 2021;25(9):3627-37. [10.1109/JBHI.2021.3066347](https://doi.org/10.1109/JBHI.2021.3066347)
- [17] Dhyani S, Kumar A, Choudhury S. Analysis of ECG-Based Arrhythmia Detection System Using Machine Learning. *MethodsX*. 2023;10:102195. [10.1016/j.mex.2023.102195](https://doi.org/10.1016/j.mex.2023.102195)
- [18] Muniasamy A, Muniasamy V, Bhatnagar R. Predictive Analytics for Cardiovascular Disease Diagnosis Using Machine Learning Techniques. In: *Advanced Machine Learning Technologies and Applications*. vol. 1141 of *Advances in Intelligent Systems and Computing*. Springer Singapore; 2021. p. 493-502. [10.1007/978-981-15-3383-9_45](https://doi.org/10.1007/978-981-15-3383-9_45)
- [19] Wang Z, Qian L, Han C, Shi L. Application of Multi-Feature Fusion and Random Forests to the Automated Detection of Myocardial Infarction. *Cognitive Systems Research*. 2020;59:15-26. [10.1016/j.cogsys.2019.09.001](https://doi.org/10.1016/j.cogsys.2019.09.001)
- [20] Hassaballah M, Wazery YM, Ibrahim IE, Farag A. ECG Heartbeat Classification Using Machine Learning and Metaheuristic Optimization for Smart Healthcare Systems. *Bioengineering*. 2023;10(4):429. [10.3390/bioengineering10040429](https://doi.org/10.3390/bioengineering10040429)
- [21] Subba T, Chingtham T. Comparative Analysis of Machine Learning Algorithms with Advanced Feature Extraction for ECG Signal Classification. *IEEE Access*. 2024;12:57727-40. [10.1109/ACCESS.2024.3387041](https://doi.org/10.1109/ACCESS.2024.3387041)
- [22] Marzog HA, Abd HJ. Machine Learning ECG Classification Using Wavelet Scattering of Feature Extraction. *Applied Computational Intelligence and Soft Computing*. 2022;2022:9884076. [10.1155/2022/9884076](https://doi.org/10.1155/2022/9884076)
- [23] Sraitih M, Jabrane Y, Hajjam El Hassani A. A Robustness Evaluation of Machine Learning Algorithms for ECG Myocardial Infarction Detection. *Journal of Clinical Medicine*. 2022;11(17):4935. [10.3390/jcm11174935](https://doi.org/10.3390/jcm11174935)
- [24] Potter EL, Rodrigues CH, Ascher DB, Abhayaratna WP, Sengupta PP, Marwick TH. Machine Learning of ECG Waveforms to Improve Selection for Testing for Asymptomatic Left Ventricular Dysfunction. *JACC: Cardiovascular Imaging*. 2021;14(10):1904-15. [10.1016/j.jcmg.2021.05.015](https://doi.org/10.1016/j.jcmg.2021.05.015)
- [25] Gomes JC, de Santana MA, Masood AI, de Lima CL, dos Santos WP. COVID-19's Influence on Cardiac Function: A Machine Learning Perspective on ECG Analysis. *Medical & Biological Engineering & Computing*. 2023;61(5):1057-81. [10.1007/s11517-023-02773-7](https://doi.org/10.1007/s11517-023-02773-7)
- [26] Kalmady SV, Salimi A, Sun W, Sepehrvand N, Nademi Y, Bailey K, et al. Development and Validation of Machine Learning Algorithms Based on Electrocardiograms for Cardiovascular Diagnoses at the Population Level. *npj Digital Medicine*. 2024;7(1):133. [10.1038/s41746-024-01130-8](https://doi.org/10.1038/s41746-024-01130-8)
- [27] Hossain AI, Sikder S, Das A, Dey A. Applying Machine Learning Classifiers on ECG Dataset for Predicting Heart Disease. In: *2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*. IEEE; 2021. p. 1-6. [10.1109/ACMI53878.2021.9528169](https://doi.org/10.1109/ACMI53878.2021.9528169)
- [28] Rahma MM, Salman AD. Heart Disease Classification-Based on the Best Machine Learning Model. *Iraqi Journal of Science*. 2022;63(9):3966-76. [10.24996/ijjs.2022.63.9.28](https://doi.org/10.24996/ijjs.2022.63.9.28)
- [29] Khan AH, Hussain M, Malik MK. ECG Images Dataset of Cardiac and COVID-19 Patients. *Data in Brief*. 2021;34:106762. [10.1016/j.dib.2021.106762](https://doi.org/10.1016/j.dib.2021.106762)

- [30] Khan AH, Hussain M. ECG Images Dataset of Cardiac Patients. Mendeley Data; 2021. Dataset, version 2. Available from: <https://data.mendeley.com/datasets/gwbz3fsgp8/2.10.17632/gwbz3fsgp8.2>
- [31] Wong M. Kaggle Datasets. Kaggle;. Dataset page; accessed 2026-08-06. Available from: <https://www.kaggle.com/datasets/morriswongch/kaggle-datasets>
- [32] Meek S, Morris F. Introduction. I—Leads, Rate, Rhythm, and Cardiac Axis. BMJ. 2002;324(7334):415-8. [10.1136/bmj.324.7334.415](https://doi.org/10.1136/bmj.324.7334.415)
- [33] Surawicz B, Knilans T. Chou's Electrocardiography in Clinical Practice: Adult and Pediatric. 6th ed. Saunders Elsevier; 2008. [10.1016/B978-1-4160-3774-3.X1000-3](https://doi.org/10.1016/B978-1-4160-3774-3.X1000-3)
- [34] Saravanan C. Color Image to Grayscale Image Conversion. In: 2010 Second International Conference on Computer Engineering and Applications. vol. 2. IEEE; 2010. p. 196-9. [10.1109/ICCEA.2010.192](https://doi.org/10.1109/ICCEA.2010.192)
- [35] Bataineh B, Tounsi M, Zamzami N, Janbi J, Abu-Ain WAK, AbuAin T, et al. A Comprehensive Review on Document Image Binarization. Journal of Imaging. 2025;11(5):133. [10.3390/jimaging11050133](https://doi.org/10.3390/jimaging11050133)
- [36] Saeid AA, Rahma AMS, Abdul Hussien MJ. Cuneiform Tablets Image Preprocessing Proposed Algorithms Techniques for Pattern Recognition. Iraqi Journal of Science. 2018;59(3A):1326-38. Available from: <https://ijs.uobaghdad.edu.iq/index.php/eijs/article/view/301>
- [37] Wang T, Yamakawa Y. Edge-Supervised Linear Object Skeletonization for High-Speed Camera. Sensors. 2023;23(12):5721. [10.3390/s23125721](https://doi.org/10.3390/s23125721)
- [38] Nasser H, Ngo P, Debled-Rennesson I. Dominant Point Detection Based on Discrete Curve Structure and Applications. Journal of Computer and System Sciences. 2018;95:177-92. [10.1016/j.jcss.2017.07.007](https://doi.org/10.1016/j.jcss.2017.07.007)
- [39] Di Ruberto C. On the Detection of Visual Features from Digital Curves Using a Meta-heuristic Approach. Applied Mathematics. 2012;3:1750-62. [10.4236/am.2012.331241](https://doi.org/10.4236/am.2012.331241)
- [40] Nguyen TP, Debled-Rennesson I. A Discrete Geometry Approach for Dominant Point Detection. Pattern Recognition. 2011;44(1):32-44. [10.1016/j.patcog.2010.06.022](https://doi.org/10.1016/j.patcog.2010.06.022)
- [41] Aguilera-Aguilera EJ, Carmona-Poyato A, Madrid-Cuevas FJ, Medina-Carnicer R. The Computation of Polygonal Approximations for 2D Contours Based on a Concavity Tree. Journal of Visual Communication and Image Representation. 2014;25(8):1905-17. [10.1016/j.jvcir.2014.09.012](https://doi.org/10.1016/j.jvcir.2014.09.012)
- [42] Tienaaah T, Stefanakis E, Coleman D. Contextual Douglas-Peucker Simplification. Geomatica. 2015;69(3):327-38. [10.5623/cig.2015-306](https://doi.org/10.5623/cig.2015-306)
- [43] Saeid AA, Ogla R, Shaker SH. A Novel Approach for Shape Pattern Recognition Based on Boundary Features Generated by Line Simplification Algorithm. Baghdad Science Journal. 2025;22(1):349-60. [10.21123/bsj.2024.9517](https://doi.org/10.21123/bsj.2024.9517)
- [44] Wu WY. Dominant Point Detection Using Adaptive Bending Value. Image and Vision Computing. 2003;21(6):517-25. [10.1016/S0262-8856\(03\)00031-3](https://doi.org/10.1016/S0262-8856(03)00031-3)
- [45] Wu ST, da Silva ACG, Márquez MR. The Douglas-Peucker Algorithm: Sufficiency Conditions for Non-Self-Intersections. Journal of the Brazilian Computer Society. 2004;9(3):67-84. [10.1590/S0104-65002004000100006](https://doi.org/10.1590/S0104-65002004000100006)
- [46] Jung JW, So BC, Kang JG, Lim DW, Son Y. Expanded Douglas-Peucker Polygonal Approximation and Opposite Angle-Based Exact Cell Decomposition for Path Planning with Curvilinear Obstacles. Applied Sciences. 2019;9(4):638. [10.3390/app9040638](https://doi.org/10.3390/app9040638)
- [47] Rosenfeld A, Johnston E. Angle Detection on Digital Curves. IEEE Transactions on Computers. 1973;C-22(9):875-8. [10.1109/TC.1973.5009188](https://doi.org/10.1109/TC.1973.5009188)

- [48] Wu SH, Yu XC, Xu M. A New Corner Detection Method Based on Differential Code. In: International Conference on Space Information Technology 2009. vol. 7651 of Proceedings of SPIE. SPIE; 2010. p. 765125. [10.1117/12.855497](https://doi.org/10.1117/12.855497)
- [49] Ehsani R, Drabløs F. Robust Distance Measures for k -NN Classification of Cancer Data. Cancer Informatics. 2020;19:1176935120965542. [10.1177/1176935120965542](https://doi.org/10.1177/1176935120965542)
- [50] Rahma AMS, Saeid AA, Hussien MJA. Recognize Assyrian Cuneiform Characters by Virtual Dataset. In: 2017 6th International Conference on Information and Communication Technology and Accessibility (ICTA). IEEE; 2017. p. 1-7. [10.1109/ICTA.2017.8336049](https://doi.org/10.1109/ICTA.2017.8336049)
- [51] Abed AM, Gitaffa SA, Issa AH. Quadratic Support Vector Machine and K -Nearest Neighbor Based Robust Sensor Fault Detection and Isolation. Engineering and Technology Journal. 2021;39(5A):859-69. [10.30684/etj.v39i5A.2002](https://doi.org/10.30684/etj.v39i5A.2002)
- [52] Nayak S, Bhat M, Reddy NS, Rao BA. Study of Distance Metrics on k -Nearest Neighbor Algorithm for Star Categorization. Journal of Physics: Conference Series. 2022;2161(1):012004. [10.1088/1742-6596/2161/1/012004](https://doi.org/10.1088/1742-6596/2161/1/012004)
- [53] Mahmood NT, Al-Muifraje MH, Salih SK, Saeed TR. Pattern Recognition of Composite Motions Based on EMG Signal via Machine Learning. Engineering and Technology Journal. 2021;39(2A):295-305. [10.30684/etj.v39i2A.1743](https://doi.org/10.30684/etj.v39i2A.1743)
- [54] Hsiao WT, Kan YC, Kuo CC, Kuo YC, Chai SK, Lin HC. Hybrid-Pattern Recognition Modeling with Arrhythmia Signal Processing for Ubiquitous Health Management. Sensors. 2022;22(2):689. [10.3390/s22020689](https://doi.org/10.3390/s22020689)
- [55] Tang K, Ma S, Sun X, Guo D. Optimizing Machine Learning for Enhanced Automated ECG Analysis in Cardiovascular Healthcare. Egyptian Informatics Journal. 2024;28:100578. [10.1016/j.eij.2024.100578](https://doi.org/10.1016/j.eij.2024.100578)
- [56] Sankar S, Potti A, Naga Chandrika G, Ramasubbareddy S. Thyroid Disease Prediction Using XGBoost Algorithms. Journal of Mobile Multimedia. 2022;18(3):917-34. [10.13052/jmm1550-4646.18322](https://doi.org/10.13052/jmm1550-4646.18322)
- [57] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM; 2016. p. 785-94. [10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785)
- [58] Jaleel HQ, Stephan JJ, Naji SA. Textual Dataset Classification Using Supervised Machine Learning Techniques. Engineering and Technology Journal. 2022;40(4):527-38. [10.30684/etj.v40i4.1970](https://doi.org/10.30684/etj.v40i4.1970)
- [59] Breiman L. Random Forests. Machine Learning. 2001;45(1):5-32. [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324)
- [60] Gislason PO, Benediktsson JA, Sveinsson JR. Random Forests for Land Cover Classification. Pattern Recognition Letters. 2006;27(4):294-300. [10.1016/j.patrec.2005.08.011](https://doi.org/10.1016/j.patrec.2005.08.011)

How to cite this article

Ali FI, Gbashi EK, Saied AA. ECG pattern recognition based on polygon approximation techniques. Journal of University of Anbar for Pure Science. 2026; 20(2):350-366. doi:[10.37652/juaps.2025.158955.1369](https://doi.org/10.37652/juaps.2025.158955.1369)